EXHIBIT 1

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendant.	)	C.A. No. 09-791-GMS
APPLE INC. Counterclaim-Plaintiff, v. NOKIA CORPORATION and NOKIA INC. Counterclaim-Defendants.	)	

DEFENDANT AND COUNTERCLAIM-PLAINTIFF APPLE INC.'S RESPONSE TO PLAINTIFF NOKIA CORPORATION'S SECOND SET OF INTERROGATORIES NOS. 6-11 (INCORRECTLY TITLED "NOKIA CORPORATION'S FIRST SET OF INTERROGATORIES (NOS. 1-6)")

Defendant and Counterclaim-Plaintiff Apple Inc. ("Apple") hereby responds to Plaintiff and Counterclaim-Defendant Nokia Corporation's ("Nokia") Second Set of Interrogatories Nos. 6-11.[1] These responses ("Responses") are based on information reasonably available to Apple at the present time. Apple reserves the right to amend and supplement the Responses when and if additional information becomes available.

[1] Nokia incorrectly titled its second set of interrogatories, served on June 25, 2010, "Nokia's First Set of Interrogatories (Nos. 1-6)." Nokia served an earlier set of interrogatories on June 10, 2010. To avoid confusion, Apple will refer to the second set of Nokia interrogatories, served on June 25, 2010, as "Nokia's Second Set of Interrogatories."

GENERAL OBJECTIONS

Each of Apple's responses is subject to and incorporates the following General Objections:

1. Apple objects to the set of interrogatories on the ground that they are incorrectly titled "Nokia Corporation's First Set of Interrogatories (Nos. 1-6)." Nokia already served a first set of interrogatories, which it numbered 1-5, on June 10, 2010.

2. Apple objects to any interrogatory to the extent that it seeks information that is protected from discovery by the attorney-client privilege, the attorney work product doctrine, the joint defense or common interest privilege, or any other applicable privilege or immunity. The inadvertent production by Apple of information protected from disclosure by any privilege or doctrine shall not constitute a waiver by Apple of such privileges or protections. To the extent any interrogatory calls for identification of attorney-client privileged or work product privileged documents or communications withheld, Apple will not identify such documents or communications created on or after October 22, 2009, as specified in clause 13 of the Protective Order.

3. To the extent an interrogatory seeks information of a confidential or proprietary nature to Apple, or to others to whom Apple is under an obligation of confidentiality, Apple will respond pursuant to the terms of the protective order entered in this case and subject to notice to third parties, as necessary.

4. Apple objects to any interrogatory to the extent it is premature and/or to the extent that it: (a) seeks information that is the subject of expert testimony; or (b) is dependent on depositions and documents that have not been taken or produced.

5. Apple objects to any interrogatory to the extent that it is vague, ambiguous, overbroad, or unduly burdensome, or purports to impose upon Apple any duty or obligation that is inconsistent with or in excess of those obligations that are imposed by the Federal Rules of Civil Procedure, the Local Rules, or any other applicable rule. In particular, Apple objects to any interrogatory to the extent that it calls for information that is neither relevant to the claims or defenses of the parties nor reasonably calculated to lead to the discovery of admissible evidence.

6. Apple objects to each interrogatory to the extent that it would impose a duty on Apple to undertake a search for or an evaluation of information, documents, or things for which Nokia is equally able to search for and evaluate. In particular, Apple objects to each interrogatory to the extent that it seeks information or documents that are publicly available.

7. Apple objects to each interrogatory to the extent that it seeks information that can be derived or ascertained from documents that will be produced in discovery or that are uniquely in Nokia's possession, custody, and control.

8. Apple objects to each interrogatory to the extent that it calls for a legal conclusion.

9. Apple objects to any definition, instruction or interrogatory to the extent that it purports to require identification of oral communications. Such definition, instruction or interrogatory is overbroad, vague, ambiguous, and unduly burdensome.

10. Apple objects to Nokia's "Instructions" and "Definitions," whether included in Nokia's Second Set of Interrogatories or incorporated by reference therein, to the extent that they attempt to impose burdens and requirements upon Apple that exceed or differ from the requirements of Rules 26, 33, and 34 of the Federal Rules of Civil Procedure and the Local Rules of the Court.

11. Apple objects to the definition of the term "Nokia" as vague, ambiguous, overbroad, and unduly burdensome to the extent that it purports to impose a duty on Apple to identify Nokia's "predecessors, successors, parents, subsidiaries …, affiliates, divisions and operating units" and "agents [or] entities under [Nokia's] control."

12. Apple objects to the definition of the terms "related to," "relate to," "relating to," or "concerning" as vague, ambiguous, overbroad, and unduly burdensome to the extent that they purport to include any meanings other than constituting, referring to, or evidencing.

13. Apple will make, and has made, reasonable efforts to respond to Nokia's Second Set of Interrogatories, to the extent that no objection is made, as Apple reasonably understands and interprets each interrogatory. If Nokia subsequently asserts any interpretation of any interrogatory that differs from the interpretation of Apple, then Apple reserves the right to supplement and amend its objections and responses.

14. All responses to the following interrogatories are based on information presently known to Apple after reasonable investigation. Discovery, fact investigation and trial preparation are ongoing, and Apple reserves its right to supplement or amend its response.

15. Nothing contained herein is an admission relative to the existence of any information sought, to the relevance or admissibility of any response, or to the truth or accuracy of any statement or characterization contained in any particular interrogatory.

16. Apple's responses to Nokia's interrogatories are made without waiver and with the preservation of: all issues as to the competency, relevancy, materiality, privilege, and admissibility of the responses and the subject matter thereof for any purpose and in any further proceeding in this lawsuit (including trial) and in any other action or matter; the right to object to the use of any such responses or the subject matter thereof on any ground in any further

proceeding in this lawsuit (including trial) and in any other action or matter; the right to object on any ground at any time to a demand or request for further response; and the right at any time to review, correct, add to, supplement, or clarify any of the responses contained herein.

17. In Apple's objections, the terms "and" and "or" are intended to be construed conjunctively or disjunctively as necessary to make the objections inclusive rather than exclusive.

18. Apple's objections as set forth herein are made without prejudice to Apple's right to assert any additional or supplemental objections pursuant to Rule 26(e).

SPECIFIC OBJECTIONS AND RESPONSES

INTERROGATORY NO. 6:

Separately for each Asserted Nokia Patent, if Apple contends that any Accused Apple Products do not infringe the patent, separately for each Accused Apple Product, state with particularity all facts on which such contentions are based, including a detailed construction of each element of each claim of the patent that Apple contends is not infringed and a specific identification of the elements or limitations of each such claim that Apple contends does not find correspondence—either literally or by the doctrine of equivalents—in the Accused Apple Product.

RESPONSE TO INTERROGATORY NO. 6:

Apple objects to Interrogatory No. 1, to the extent it seeks the identity of "all" facts supporting Apple's contentions, on the grounds that it is overly broad and unduly burdensome. Apple further objects to the interrogatory on the grounds that it is premature, and calls for expert opinions. Nokia has not yet identified the claims it intends to assert or articulated the basis for its contention that the Accused Apple Products infringe those claims, and the Court has not yet construed the claims. Moreover, the Court's Scheduling Order does not contemplate the disclosure of claim construction positions until March 15, 2011, or rebuttal expert reports (where Apple's retained experts will provide opinions on non-infringement) until September 15, 2011.

Apple further objects to the interrogatory to the extent it requires information outside Apple's possession, custody, and control, including, for example, information concerning the technical details of components that Apple has purchased from third parties.

Subject to and without waiving the foregoing General and Specific Objections, Apple believes that the parties should disclose the basis for their patent-related contentions, but that this disclosure should take place in a logical and orderly manner. In particular, Nokia should disclose the patent claims it intends to assert, and the basis for its belief that the Accused Apple Products are infringing those claims, before Apple discloses its non-infringement contentions with respect to those claims. Apple is willing to meet and confer regarding a reasonable schedule for the exchange of disclosures on the patent issues.

INTERROGATORY NO. 7:

Separately for each Asserted Nokia Patent, state what Apple contends is the level of ordinary skill in the art to which the inventions claimed in the patent pertain.

RESPONSE TO INTERROGATORY NO. 7:

Apple objects to Interrogatory No. 2 on the ground that it is premature, and calls for expert opinions. Subject to and without waiving the foregoing General and Specific Objections, Apple's experts will provide opinions on the level of ordinary skill in the art in their expert reports.

INTERROGATORY NO. 8:

Separately for each Asserted Nokia Patent, identify all prior art, if any, that Apple contends renders any claim of the patent unpatentable under 35 U.S.C. § 102 or 35 U.S.C. § 103, and state with particularity the alleged correspondence between each such prior art reference and each element of each such claim of the patent and, to the extent any references must be combined, how the references are combined and what teaches, motivates, or suggests the combination.

RESPONSE TO INTERROGATORY NO. 8:

Apple objects to Interrogatory No. 3 on the grounds that it is overly broad and unduly burdensome. Apple further objects to the interrogatory, to the extent it requests a complete articulation of all of Apple's contentions under § 102 and § 103, on the grounds that it is premature, and calls for expert opinions. Nokia has not yet identified the claims it intends to assert or articulated the basis for its contention that the Accused Apple Products infringe those claims, and the Court has not yet construed the claims. Moreover, the Court's Scheduling Order does not contemplate the disclosure of claim construction positions until March 15, 2011, or initial expert reports (where Apple's retained experts will provide opinions on invalidity) until August 15, 2011.

Subject to and without waiving the foregoing General and Specific Objections, as stated above, Apple believes that the parties should disclose the basis for their patent-related contentions, but that this disclosure should take place in a logical and orderly manner. Apple is willing to meet and confer regarding a reasonable schedule for the exchange of disclosures on the patent issues.

INTERROGATORY NO. 9:

Separately for each Asserted Nokia Patent, if Apple contends that the patent fails to meet the requirements of 35 U.S.C. § 101 or 35 U.S.C. § 112, state with particularity all facts on which such contentions are based and identify all documents that relate to, support, or detract from the basis for such contentions.

RESPONSE TO INTERROGATORY NO. 9:

Apple objects to Interrogatory No. 4 on the grounds that it is overly broad and unduly burdensome. Apple further objects to the interrogatory, to the extent it requests a complete articulation of all of Apple's contentions under § 101 and § 112, on the grounds that it is premature, and calls for expert opinions. Nokia has not yet identified the claims it intends to

assert or articulated the basis for its contention that the Accused Apple Products infringe those claims, and the Court has not yet construed the claims. Moreover, the Court's Scheduling Order does not contemplate the disclosure of claim construction positions until March 15, 2011, or initial expert reports (where Apple's retained experts will provide opinions on invalidity) until August 15, 2011.

Subject to and without waiving the foregoing General and Specific Objections, as stated above, Apple believes that the parties should disclose the basis for their patent-related contentions, but that this disclosure should take place in a logical and orderly manner. Apple is willing to meet and confer regarding a reasonable schedule for the exchange of disclosures on the patent issues.

INTERROGATORY NO. 10:

Separately for each Asserted Nokia Patent, if Apple contends that the patent is unenforceable, state with particularity all facts on which such contentions are based, identify all persons with knowledge concerning the basis for such contentions, and identify all documents that relate to, support, or detract from the basis for such contentions.

RESPONSE TO INTERROGATORY NO. 10:

Apple objects to Interrogatory No. 5 on the grounds that it is overly broad and unduly burdensome. Apple further objects to the interrogatory, to the extent it requests a complete articulation of all of Apple's unenforceability contentions, on the grounds that it is premature, and calls for expert opinions. Discovery is ongoing.

Subject to and without waiving the foregoing General and Specific Objections, the Nokia Asserted Patents are unenforceable against Apple because Nokia's misconduct before standards-setting organizations, as set forth below and more fully in Apple's Counterclaim, estops Nokia from enforcing and waives Nokia's right to enforce the Asserted Nokia Patents. Nokia's deliberate and deceptive non-disclosure of purported essential IPR during the standards-setting

process, its false representations to SSOs that it would license on F/RAND terms the patents it belatedly declared essential, and Nokia's assertion of its wrongfully obtained monopoly power against Apple in demanding non-F/RAND license terms from Apple and then suing Apple for patent infringement when Apple refused to accede to those unreasonable demands (even though Apple had a license as a matter of law), also constitutes patent misuse and renders the Nokia Asserted Patents unenforceable. In addition, Nokia's demand for a reciprocal "grantback" license to Apple's non-standards-essential patents as a condition for licensing the Asserted Nokia Patents at a F/RAND royalty rate constitutes misuse of the Asserted Nokia Patents, rendering them unenforceable.

INTERROGATORY NO. 11:

Separately for each Asserted Nokia Patent, identify the highest royalty rate that Apple believes constitutes a FRAND rate.

RESPONSE TO INTERROGATORY NO. 11:

Apple objects to Interrogatory No. 6, on the grounds that it is premature, and calls for expert opinions. The information Nokia seeks will be the subject of fact discovery, including discovery from Nokia concerning its claim that each Nokia Asserted Patent is essential to a standard covering Apple's products, discovery of any licenses Nokia has with third parties covering each Asserted Nokia Patent, and discovery from third parties concerning any licenses with Nokia covering each Asserted Nokia Patent. Moreover, the Court's Scheduling Order does not contemplate the disclosure of expert reports until August 15, 2011.

Subject to the foregoing General and Specific Objections, Apple's experts will provide opinions on FRAND royalty rates in their expert reports.

OF COUNSEL:

William F. Lee
WILMERHALE
60 State Street
Boston, MA 02109
Tel: 617 526 6000

Mark D. Selwyn
WILMERHALE
1117 California Avenue
Palo Alto, CA 94304
Tel: (650) 858-6000

Kenneth H. Bridges
Michael T. Pieja
WONG CABELLO
540 Cowper Street, Suite 100
Palo Alto, CA
Tel: (650) 681-4475

Dated: July 26, 2010
975957 / 35035

POTTER ANDERSON & CORROON LLP

By: */s/ David E. Moore*

Richard L. Horwitz (#2246)
David E. Moore (#3983)
Hercules Plaza, 6th Floor
1313 N. Market Street
Wilmington, DE 19899
Tel: (302) 984-6000
rhorwitz@potteranderson.com
dmoore@potteranderson.com

Attorneys for Defendant/Counterclaim-Plaintiff Apple Inc.

CERTIFICATE OF SERVICE

I, David E. Moore, hereby certify that on July 26, 2010, true and correct copies of the within document were served on the following counsel of record at the addresses and in the manner indicated:

VIA ELECTRONIC MAIL

Jack B. Blumenfeld
Rodger D. Smith II
Morris, Nichols, Arsht & Tunnell LLP
1201 North Market Street
Wilmington, DE 19899
jblumenfeld@mnat.com
rsmith@mnat.com

Adam J. Biegel
Keith E. Broyles
Patrick J. Flinn
John D. Haynes
Ryan W. Koppelman
Mark A. McCarty
Peter Kontio
Alston & Bird LLP
One Atlantic Center
1201 West Peachtree Street
Atlanta, GA 30309
adam.biegel@alston.com
keith.broyles@alston.com
patrick.flinn@alston.com
john.haynes@alston.com
ryan.koppelman@alston.com
mark.mccarty@alston.com
peter.kontio@alston.com

Alan L. Whitehurst
The Atlantic Building
950 F Street, NW
Washington, DC 20004-1404
alan.whitehurst@alston.com

/s/ David E. Moore
David E. Moore

969800 / 35035

EXHIBIT 2

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

NOKIA CORPORATION, Plaintiffs, v. APPLE INC., Defendants.	)	C.A. No. 09-791 GMS
NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendants.	)	C.A. No. 09-1002 GMS

Caption continued on next page

APPLE INC. AND NEXT SOFTWARE, INC.'S MOTION FOR CONSOLIDATION OF THE CAPTIONED CASES FOR THE PURPOSE OF COORDINATING PRETRIAL PROCEEDINGS

Richard K. Herrmann (No. 405)
Mary B. Matterer (I.D. #2696)
MORRIS JAMES, LLP
500 Delaware Ave., Suite 1500
Wilmington, DE 19801
(302) 888-6800
rherrmann@morrisjames.com

Attorneys for APPLE INC.
and NEXT SOFTWARE, INC.

May 24, 2010

APPLE INC., and NeXT SOFTWARE, INC., f/k/a NeXT COMPUTER, INC., Plaintiffs, v. HIGH TECH COMPUTER CORP., a/k/a HTC CORP., HTC (B.V.I.) CORP., HTC AMERICA, INC., and EXEDEA, INC., Defendants.	)	C.A. No. 10-166-RK
APPLE INC., Plaintiff, v. HIGH TECH COMPUTER CORP., a/k/a HTC CORP., HTC (B.V.I.) CORP., HTC AMERICA, INC., and EXEDEA, INC., Defendants.	)	C.A. No. 10-167-RK

Apple Inc. and NeXT Software, Inc. (collectively, "Apple") hereby move this Court for an order, pursuant to Federal Rule of Civil Procedure 42(a), consolidating the captioned cases before Chief Judge Sleet for all pretrial proceedings and common issues of law and fact relating to fact discovery for the captioned cases. The grounds for this motion are set forth in Apple Inc. and NeXT Software, Inc.'s Brief in Support of Their Motion for Consolidation of the Captioned Cases for the Purpose of Coordinating Pretrial Proceedings and in the Declaration of Richard K. Herrmann filed contemporaneously herewith.

Dated: May 24, 2010

/s/ Richard K. Herrmann

Richard K. Herrmann (I.D. #405)
Mary B. Matterer (I.D. #2696)
MORRIS JAMES LLP
500 Delaware Avenue, Suite 1500
Wilmington, Delaware 19801
Telephone: (302) 888-6800
rherrmann@morrisjames.com

Robert G. Krupka, P.C.
KIRKLAND & ELLIS LLP
333 South Hope Street
Los Angeles, California 90071
Telephone: (213) 680-8400
Facsimile: (213) 680-8500

Gregory S. Arovas, P.C.
KIRKLAND & ELLIS LLP
601 Lexington Avenue
New York, New York 10022
Telephone: (212) 446-4800
Facsimile: (212) 446-4900

Bryan S. Hales, P.C.
Marcus E. Sernel, P.C.
KIRKLAND & ELLIS LLP
300 North LaSalle
Chicago, IL 60654
Telephone: (312) 862-2000
Facsimile: (312) 862-2200

Kenneth H. Bridges
Michael T. Pieja
Brian C. Kwok
WONG, CABELLO, LUTSCH,
RUTHERFORD & BRUCCULERI LLP
540 Cowper Street, Suite 100
Palo Alto, CA 94301
Telephone: (650) 681-4475
Facsimile: (650) 403-4043

Attorneys for Apple Inc. and NeXT Software, Inc. f/k/a NeXT Computer, Inc.

RULE 7.1.1 STATEMENT

Counsel for Apple Inc. and NeXT Software, Inc. hereby states that it contacted opposing counsel in an effort to resolve the issues raised in its Motion to Consolidate, but to no avail. Nokia has not responded to Apple's inquiry and HTC confirmed that it will oppose consolidating the present cases.

Dated: May 24, 2010

/s/ Richard K. Herrmann
Richard K. Herrmann

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendant.	)	C.A. No. 09-791 GMS
NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendant.	)	C.A. No. 09-1002 GMS
APPLE INC., and NeXT SOFTWARE, INC., f/k/a NeXT COMPUTER, INC., Plaintiffs, v. HIGH TECH COMPUTER CORP., a/k/a HTC CORP., HTC (B.V.I.) CORP., HTC AMERICA, INC., and EXEDEA, INC., Defendants.	)	C.A. No. 10-166-RK
APPLE INC., Plaintiff, v. HIGH TECH COMPUTER CORP., a/k/a HTC CORP., HTC (B.V.I.) CORP., HTC AMERICA, INC., and EXEDEA, INC., Defendants.	)	C.A. No. 10-167-RK

[PROPOSED] ORDER

Having considered Apple Inc. and NeXT Software, Inc.'s Motion for Consolidation and the related briefing,

IT IS HEREBY ORDERED this _____ day of _____, 2010 that the Motion is GRANTED. The cases captioned as Nokia Corporation v. Apple Inc. (C.A. Nos. 09-791-GMS and 09-1002-GMS), Apple Inc. and NeXT Software, Inc. v. High Tech Computer Corp., et al., (C.A. No. 10-166-RK) and Apple Inc. v. High Tech Computer Corp., et al. (C.A. No. 10-167-RK) are hereby consolidated for all pretrial proceedings and common issues of law and fact relating to fact discovery.

_____

Gregory M. Sleet, Chief Judge

Motions

1:09-cv-00791-GMS Nokia Corporation v. Apple Inc.
PATENT

U.S. District Court

District of Delaware

Notice of Electronic Filing

The following transaction was entered by Herrmann, Richard on 5/24/2010 at 8:34 PM EDT and filed on 5/24/2010

Case Name: Nokia Corporation v. Apple Inc.
Case Number: 1:09-cv-00791-GMS
Filer: Apple Inc.
Document Number: 47

Docket Text:
MOTION to Consolidate Cases *09-791; 09-1002; 10-166; and 10-167* - filed by Apple Inc.. (Herrmann, Richard)

1:09-cv-00791-GMS Notice has been electronically mailed to:

David Ellis Moore dmoore@potteranderson.com, bill.rodgers@mylan.com, ntarantino@potteranderson.com, shamlin@potteranderson.com

Jack B. Blumenfeld jbbefiling@mnat.com

John D Haynes john.haynes@alston.com

Kenneth H. Bridges kbridges@wongcabello.com

Mark D. Selwyn Mark.Selwyn@wilmerhale.com

Michael T. Pieja mpieja@wongcabello.com

Patrick J. Flinn pat.flinn@alston.com

Richard K. Herrmann rherrmann@morrisjames.com, shadley@morrisjames.com, tsmiley@morrisjames.com

Richard L. Horwitz rhorwitz@potteranderson.com, iplitigation@potteranderson.com, mbaker@potteranderson.com, nmcmenamin@potteranderson.com, shamlin@potteranderson.com

Rodger Dallery Smith , II rdsefiling@mnat.com

Ryan W. Koppelman ryan.koppelman@alston.com

William F. Lee William.Lee@wilmerhale.com

1:09-cv-00791-GMS Notice has been delivered by other means to:

The following document(s) are associated with this transaction:

Document description: Main Document
Original filename:n/a
Electronic document Stamp:
[STAMP dcecfStamp_ID=1079733196 [Date=5/24/2010] [FileNumber=987828-0]
[0aecc576a1b3d1bbe227275653b8a3ec1208a4849335870e7bd31d35d7d7f93c8fc8
73bc288b5c63385bc3faeb14a964bef31f26c75db0b25149cf03f0f4189f]]

Motions

1:09-cv-01002-GMS Nokia Corporation v. Apple Inc.
STAYED, PATENT

U.S. District Court

District of Delaware

Notice of Electronic Filing

The following transaction was entered by Herrmann, Richard on 5/24/2010 at 8:35 PM EDT and filed on 5/24/2010

Case Name: Nokia Corporation v. Apple Inc.
Case Number: 1:09-cv-01002-GMS
Filer: Apple Inc.
Document Number: 17

Docket Text:
MOTION to Consolidate Cases *09-791; 09-1002; 10-166; and 10-167* - filed by Apple Inc.. (Herrmann, Richard)

1:09-cv-01002-GMS Notice has been electronically mailed to:

David Ellis Moore dmoore@potteranderson.com, bill.rodgers@mylan.com, ntarantino@potteranderson.com, shamlin@potteranderson.com

Jack B. Blumenfeld jbbefiling@mnat.com

Kenneth H. Bridges kbridges@wongcabello.com

Mark D. Selwyn Mark.Selwyn@wilmerhale.com

Michael T. Pieja mpieja@wongcabello.com

Richard K. Herrmann rherrmann@morrisjames.com, shadley@morrisjames.com, tsmiley@morrisjames.com

Richard L. Horwitz rhorwitz@potteranderson.com, iplitigation@potteranderson.com, mbaker@potteranderson.com, nmcmenamin@potteranderson.com, shamlin@potteranderson.com

Rodger Dallery Smith , II rdsefiling@mnat.com

William F. Lee William.Lee@wilmerhale.com

1:09-cv-01002-GMS Notice has been delivered by other means to:

The following document(s) are associated with this transaction:

Document description: Main Document
Original filename:n/a
Electronic document Stamp:
[STAMP dcecfStamp_ID=1079733196 [Date=5/24/2010] [FileNumber=987831-0]
[80ed1442982979f0dd1a831a2013553a571791c67f6e652f63f3edb075d7eb3d8989
43ffbcbad2250fc94ba86747beb26353fd54d3b77bd8adca92a6d65513b1]]

Motions

1:10-cv-00166-RK Apple Inc. et al v. High Tech Computer Corp. et al
STAYED, INTRACIRCUIT, PATENT, VACANTJUDGESHIP

U.S. District Court

District of Delaware

Notice of Electronic Filing

The following transaction was entered by Herrmann, Richard on 5/24/2010 at 8:37 PM EDT and filed on 5/24/2010

Case Name: Apple Inc. et al v. High Tech Computer Corp. et al
Case Number: 1:10-cv-00166-RK
Filer: Apple Inc.
Document Number: 21

Docket Text:
MOTION to Consolidate Cases *09-791; 09-1002; 10-166; and 10-167* - filed by Apple Inc.. (Herrmann, Richard)

1:10-cv-00166-RK Notice has been electronically mailed to:

Karen L. Pascale kpascale@ycst.com, corpcal@ycst.com, corporate@ycst.com

Richard K. Herrmann rherrmann@morrisjames.com, shadley@morrisjames.com, tsmiley@morrisjames.com

1:10-cv-00166-RK Notice has been delivered by other means to:

The following document(s) are associated with this transaction:

Document description: Main Document
Original filename:n/a
Electronic document Stamp:
[STAMP dcecfStamp_ID=1079733196 [Date=5/24/2010] [FileNumber=987834-0] [afcdf415543664703d84d6ce20e3e7c02dd536a70d4126a4c5f1525f070bbb3dfb43640f4fa385ccb277434906c3af0d19e26b71db6365b21f580e3d1072acf8]]

Motions

1:10-cv-00167-RK Apple Inc. v. High Tech Computer Corp. et al
INTRACIRCUIT, PATENT, VACANTJUDGESHIP

U.S. District Court

District of Delaware

Notice of Electronic Filing

The following transaction was entered by Herrmann, Richard on 5/24/2010 at 8:38 PM EDT and filed on 5/24/2010

Case Name: Apple Inc. v. High Tech Computer Corp. et al
Case Number: 1:10-cv-00167-RK
Filer: Apple Inc.
Document Number: 19

Docket Text:
MOTION to Consolidate Cases *09-791; 09-1002; 10-166; and 10-167* - filed by Apple Inc.. (Herrmann, Richard)

1:10-cv-00167-RK Notice has been electronically mailed to:

Karen L. Pascale kpascale@ycst.com, corpcal@ycst.com, corporate@ycst.com

Richard K. Herrmann rherrmann@morrisjames.com, shadley@morrisjames.com, tsmiley@morrisjames.com

1:10-cv-00167-RK Notice has been delivered by other means to:

The following document(s) are associated with this transaction:

Document description: Main Document
Original filename:n/a
Electronic document Stamp:
[STAMP dcecfStamp_ID=1079733196 [Date=5/24/2010] [FileNumber=987837-0] [ccb8b07f436e0b176309a791e8bfeb03bd15e5b3531e7fbd00e79a9a339b01034cb1d43e77ab240dfa38b3fb34a4afab61b118c5c801c9b9dcff6367d45356ba]]

EXHIBIT 3

IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF WISCONSIN

NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendant.	CIVIL ACTION NO. 10-CV-249
APPLE INC., Counterclaim-Plaintiff, v. NOKIA CORPORATION and NOKIA INC., Counterclaim-Defendants.	**JURY TRIAL DEMANDED**

APPLE INC.'S MOTION TO TRANSFER VENUE
TO THE DISTRICT OF DELAWARE PURSUANT TO 28 U.S.C. § 1404(a)

Defendant Apple Inc. hereby moves this Court pursuant to 28 U.S.C. § 1404(a) to transfer this action to the District of Delaware. The grounds for this motion are set forth in Apple Inc.'s Memorandum In Support Of Motion to Transfer Venue To The District Of Delaware Pursuant to 28 U.S.C. § 1404(a) and in the Declarations of Mark Selwyn and Mark Bentley filed contemporaneously herewith.

GODFREY & KAHN, S.C.

OF COUNSEL:

WILMER CUTLER PICKERING
HALE AND DORR LLP

William F. Lee
Cynthia D. Vreeland
Dominica E. Massa
60 State Street, Boston, MA 02109
Tel. (617) 526-6000

Keith Slenkovich
1117 California Avenue
Palo Alto, CA 94304
Tel. (650) 858-6000

Dated: June 29, 2010

By: *s/James D. Peterson*

James Donald Peterson (# 1022819)
One East Main Street, Suite 500
P.O. Box 2719
Madison, WI 53701-2719
Tel: (608) 257-3911
Fax: (608) 257-0609
jpeterson@gklaw.com

Attorneys for Defendant and Counterclaim-Plaintiff Apple Inc.

CERTIFICATE OF SERVICE

I hereby certify that on June 29, 2010, I caused these documents to be electronically filed with the Clerk of Court using the ECF system,

- Apple Inc.'s Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a);
- Apple Inc.'s Memorandum in Support of Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a);
- Declaration of Mark Selwyn in Support of Defendant's Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a) with Exhibits 1 - 14; and
- Declaration of Mark Bentley in Support of Defendant's Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a)

which will make these documents available to all counsel of record for viewing and downloading from the ECF system.

The undersigned further certifies that manual notice was sent to the following non-CM/ECF participant by email:

John D. Haynes Alston & Bird, LLP 1201 W. Peachtree Street Atlanta, GA 30309-3424 Tel: 404-881-7000 Fax: 404-881-7777 *Attorneys for Nokia Corporation*	john.haynes@alston.com

s/James D. Peterson
James D. Peterson

5137299_2

IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF WISCONSIN

NOKIA CORPORATION, Plaintiff, v. APPLE INC., Defendant.	CIVIL ACTION NO. 10-CV-249
APPLE INC., Counterclaim-Plaintiff, v. NOKIA CORPORATION and NOKIA INC., Counterclaim-Defendants.	**JURY TRIAL DEMANDED**

APPLE INC.'S MEMORANDUM IN SUPPORT OF MOTION TO TRANSFER VENUE TO THE DISTRICT OF DELAWARE PURSUANT TO 28 U.S.C. § 1404(a)

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Federal Cases

Federal Statutes

I. Nature And Stage Of The Proceedings

Plaintiff Nokia Corporation ("Nokia") filed its Complaint for Patent Infringement against Defendant Apple Inc. ("Apple") on May 7, 2010 (Docket No. 1). Apple filed its Answer, Defenses, and Counterclaims on June 28, 2010.

II. Preliminary Statement

This case arises out of a patent and licensing dispute between Nokia and Apple. The same overarching dispute is already the subject of two pending lawsuits that Nokia initiated in the U.S. District Court for the District of Delaware – the district Nokia chose as the forum for the parties' larger dispute. Because all three cases arise out of a single business dispute between the same parties, about the same products, all three cases should be litigated in the same venue. Apple accordingly submits this Motion to Transfer under 28 U.S.C. § 1404(a). In this case, the District of Delaware—the district Nokia chose for its two other pending patent infringement cases against Apple—is the venue consistent with the convenience of the parties, the convenience of the witnesses and the interests of justice.

Moreover, this case has no meaningful connection to the Western District of Wisconsin. As Nokia itself asserted in 2007 in seeking to transfer a case out of this district—and as is still true today—Nokia does not have "any connection to this district or the state of Wisconsin." (Ex. 1, Nokia Reply In Support Of Motion to Transfer at 1, *Qualcomm Inc. v. Nokia Corp. et al.*, No. 3:07-CV-0187 (W.D. Wis., June 25, 2007) (emphasis in original). Exhibits referenced in this memorandum are attached to the Declaration of Mark Selwyn in Support of Apple Inc.'s Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a).) Likewise, Apple does not maintain any relevant offices in Wisconsin; it has no relevant employees in Wisconsin;

and it has no relevant documents or other evidence in Wisconsin.[1] (Declaration of Mark Bentley in Support of Apple Inc.'s Motion to Transfer Venue to the District of Delaware Pursuant to 28 U.S.C. § 1404(a) ("Bentley Decl.") at ¶ 7.)

There can be no genuine dispute, especially in light of Nokia's previously-filed litigation in the District of Delaware, that the District of Delaware is a clearly more convenient forum for the parties and the witnesses (including the overlapping third-party witnesses). Transfer would likewise serve the interests of justice by avoiding duplicative litigation and potentially conflicting rulings, and by facilitating consolidation of these related cases. As the Supreme Court has made clear, litigating the same issues in multiple cases across different district courts inevitably "leads to the wastefulness of time, energy and money that [28 U.S.C.] § 1404(a) was designed to prevent." *Continental Grain Co. v. The Barge FBL-585*, 364 U.S. 19, 26 (1960).

For these reasons, as set forth more fully below, Apple respectfully requests that this Court transfer the present action to the U.S. District Court for the District of Delaware.

III. Factual Background

A. The Related Pending Litigation In The District Of Delaware

This case is part of an ongoing business dispute between Nokia and Apple about the technology used in the parties' wireless communication devices, namely Apple's iPhone, iPhone 3G, iPhone 3GS, and recently launched iPad 3G products, and Nokia's E71, N97, N900, N8 and related products. The dispute arose from failed licensing negotiations between Nokia and Apple, in which Nokia attempted to improperly leverage certain of its patents in an effort to obtain a

[1] Three of Apple's more than 220 retail stores are located in Wisconsin. (Bentley Decl. at ¶ 7.) The Apple retail stores in Wisconsin are engaged only in general sales, service, and marketing activities. *Id.*

license to the highly valuable Apple technology used in Apple's iPhone products. As one Nokia executive candidly conceded, with the launch of the iPhone, "the market changed suddenly and [Nokia was] not fast enough changing with it." (Ex. 2, Abhinav Ramnarayan, "Nokia Fights Back For Share Of Smartphone Market," The Guardian (London), Sept. 2, 2009, at 1.)

When Apple rejected Nokia's demands, Nokia began filing a series of complaints claiming that Apple's products infringe Nokia's patents. In the first case, filed on October 22, 2009, in the District of Delaware, Nokia alleged infringement of ten patents based on "wireless communication devices such as the Apple iPhone, the Apple iPhone 3G, and the Apple iPhone 3GS." (*See* Ex. 3, Compl. ¶ 70, *Nokia Corp. v. Apple Inc.*, No. 09-CV-791 (D. Del. Oct. 22, 2009) ("Delaware I" or the "791 Case").) Apple filed counterclaims for breach of contract and attempted monopolization (based on Nokia's promotion of standards to the relevant standards-setting organizations while concealing its own patent applications allegedly covering those standards, as well as its assertion of patents it was legally obligated to license on fair, reasonable, and non-discriminatory terms), as well as infringement of nine Apple patents by Nokia's N900, as well as other related products. (*See* Ex. 4, Apple's First Am. Answer, Defenses, and Countercls. ¶¶ 201, 207, 213, 219, 225, 231, 237, 243, 249, *Nokia Corp. v. Apple Inc.*, No. 09-CV-791 (D. Del. Feb. 19, 2010).)

On December 11, 2009, Nokia filed a second complaint in the District of Delaware, alleging infringement of seven additional patents based on Apple's iPhone 3G and iPhone 3GS products. (*See* Ex. 5, Compl. ¶ 9, *Nokia Corp. v. Apple Inc.*, No. 09-CV-1002 (D. Del. Dec. 29, 2009) ("Delaware II" or the "1002 Case").) Nokia also raised the same claims in a related complaint in the U.S. International Trade Commission ("I.T.C."). (*See* Ex. 6, Compl., *In re Certain Mobile Communications and Computer Devices and Components Thereof*, Investigation

No. 337-TA-701 (Dec. 29, 2009).)[2] Apple responded to Nokia's second Delaware complaint with counterclaims for infringement of nine Apple patents, again based on Nokia's N900, as well as other related products. (*See* Ex. 7, Apple's Answer, Defenses, and Countercls. ¶¶ 52, 76, *Nokia Corp. v. Apple Inc.*, No. 09-CV-1002 (D. Del. Feb. 24, 2010).) Apple also filed a related Complaint in the I.T.C. alleging infringement of the same nine Apple patents. (*See* Ex. 8, Compl., *In re Certain Mobile Communications and Computer Devices and Components Thereof*, Investigation No. 337-TA-704 (Jan. 15, 2010).)

Nokia filed this third lawsuit, alleging infringement of five additional patents, just four days after the Delaware court entered its scheduling order in the first Delaware case—an order that permits Nokia to amend its complaint through August 30, 2010. (Ex. 9, Docket containing Scheduling Order, Delaware I, May 3, 2010.) Nokia's complaint focuses on the same Apple products: the iPhone, iPhone 3G and iPhone 3GS, as well as, for one of the five patents, Apple's iPad. Compl. ¶ 24. Apple responded to Nokia's third complaint with counterclaims for infringement of seven additional Apple patents, based on Nokia's N97, N900, and N8, and related mobile communication products. (Answer, Defenses, and Countercls. ¶¶ 39, 44, 49, 54, 59, 69.)

There can be no question that this case is part of the same dispute as Nokia's two Delaware cases. The cases involve the same parties, the same technology, and the same products, and will involve numerous common issues of law and fact. By way of example only:

[2] Nokia and Apple claimed infringement of the same patents in the second Delaware action and the I.T.C. Because the I.T.C. is authorized only to issue exclusion orders, and cannot award damages, it is a common practice for parties to initiate simultaneous actions in the I.T.C. and a federal district court.

- The same products are at issue in all three cases. Nokia has accused the same Apple products—the iPhone, iPhone 3G, and iPhone 3GS—of infringement in all three suits. Apple has accused Nokia's N97, N900, N8, and other related products.
- The same technology will be at the center of all three cases. The parties' claims for infringement will require the Court and the jury to understand the manner in which mobile devices interface with users, and transmit and receive user information over the air, including the manner in which these devices encode, modulate, and encrypt the information transmitted over the air, and send and receive related control information. (Compl. ¶¶ 14-23.)
- Indeed, the cases involve not only the same technology, but some of the very same components. For example, Nokia has included allegations focused on Apple's antennas in both this case and in Delaware. (Compl. ¶¶ 18-23, 29-39, 45-49; Ex. 5, Delaware II Compl. ¶¶ 19-27.) In fact, three of the five patents Nokia has asserted in this case accuse Apple's antennas, the same antennas accused in Delaware.
- Similarly, the modulator that is one of the focuses of Nokia's allegations here (Compl. ¶¶ 14-15, 40-44) is embedded on the same microchip, manufactured by the same third-party (Infineon Technologies), as the voltage control oscillator ("VCO") that is one of the focuses of Nokia's allegations in Delaware. (*See* Ex. 5, Delaware II Compl. ¶¶ 10-18.)
- The user interfaces on the parties' accused phones also will be a focus both of this case and in Delaware. Four of the patents Nokia has asserted in Delaware, and two of the patents Apple has asserted in Delaware, relate to the parties' user interfaces. (Ex. 5, Delaware II Compl. ¶¶ 35-72; Ex. 4, Delaware I First Am. Answer, Defense, and Countercls. ¶¶ 211-216, 229-234.) Two of the patents Apple has asserted in this case also focus on the user interface. (Answer, Defenses and Countercls. ¶¶ 24-27, 38-47.)
- In addition, device interfaces on the parties' phones will be a focus both of this case and in Delaware. One of the patents Apple has asserted in Delaware and two of the patents that it has asserted here related to device interfaces. (Ex. 4, Delaware I First Am. Answer, Defenses, and Countercls. ¶¶ 199-204; Answer, Defenses, and Countercls. ¶¶ 30-33, 48-57.)
- Furthermore, several of the patents Apple has asserted in Delaware, and one of the patents it has asserted here, relate to object oriented operating systems. (Ex. 4, Delaware I First Am. Answer, Defenses, and Countercls. ¶¶ 217-222, 235-240; Answer, Defenses, and Countercls. ¶¶ 34-35, 63-67.)
- The same financial information also will be relevant to all three cases. The parties' respective claims for damages will require discovery of the same product sales information, the same documents and, in all likelihood, testimony from the same marketing and finance employees at Apple and Nokia.
- The same third-party manufacturers are likely to be subpoenaed for documents and testimony in this case as well as in Delaware. These third parties include, for example, Infineon Technologies, Foxconn Electronics Inc., and Samsung Electronics America, Inc.

- The same prosecuting attorneys are also likely to be subpoenaed in all three cases. For example, the same lawyers at Perman & Green LLP prosecuted three of the patents Nokia has asserted in Wisconsin (U.S. Patent Nos. 6,373,345; 7,558,696; and 6,603,431) (Compl. at Exs. A, B, E), as well as six of the patents Nokia has asserted in Delaware (U.S. Patent Nos. 5,802,465; 6,359,904; 6,694,135; 5,946,651; 6,882,727; and 6,262,735) (Ex. 3, Delaware I Compl. at Exs. A, B, F, G, H; Ex. 5, Delaware II Compl. at Ex. F.)

Moreover, in addition to these substantial overlaps with the litigation between Apple and Nokia in Delaware, the case also has significant overlaps with three pending lawsuits between Apple and High Tech Computer Corp. and its subsidiaries (collectively, "HTC") in Delaware.[3] Indeed, three of the patents that Apple has asserted in this case have also been asserted against HTC in Delaware.

Given the substantial overlaps in the pending cases between Nokia, Apple, and HTC, Apple has moved to consolidate the two Nokia cases (791 and 1002 Cases) and the two HTC cases (166 and 167 Cases) currently pending in Delaware, and will move to consolidate this case as well, if the Court agrees to transfer the case to Delaware.[4] (*See* Ex. 10, Apple Inc. and Next Software, Inc.'s Mot. For Consolidation, *Nokia Corp. v. Apple Inc.*, No. 09-CV-791 (D. Del. May 24, 2010); Ex. 11, Apple Inc. and Next Software, Inc.'s Mot. For Consolidation, *Nokia Corp. v. Apple Inc.*, No. 09-CV-1002 (D. Del. May 24, 2010); Ex. 12, Apple Inc. and Next Software, Inc.'s Mot. For Consolidation, *Apple Inc. et al. v. High Tech Computer Corp. et al.*, No. 10-CV-166 (D. Del. May 24, 2010); Ex. 13, Apple Inc. and Next Software, Inc.'s Mot. For Consolidation, *Apple Inc. et al. v. High Tech Computer Corp. et al.*, No. 10-CV-167, (D. Del. May 24, 2010).) Consolidation is not only appropriate, but also could be easily accomplished

[3] The three cases are *Apple Inc. et al. v. High Tech Computer Corp. et al.*, No. 10-CV-166 (the "166 Case"), *Apple Inc. v. High Tech Computer Corp. et al.*, No. 10-CV-167 (the "167 Case"), and *Apple Inc. v. High Tech Computer Corp. et al.*, No. 10-CV-544 (the "544 Case").

[4] The 544 Case was filed early last week on June 21, 2010. Apple intends to add the 544 Case to the pending motions to consolidate.

because all six cases are in the early stages of litigation.[5] The court issued its scheduling order in the first Delaware action on May 3, 2010, and the schedule negotiated and jointly submitted by the parties in that case leaves ample time to incorporate Nokia's allegations in this case—discovery will not close until July 11, 2011. (*See* Ex. 9, Docket containing Scheduling Order, Delaware I, May 3, 2010.)

B. The Parties' Relationship To The Districts Of Delaware And Wisconsin

1. Neither Nokia Nor Apple Have Any Connection To Wisconsin

Neither Nokia nor Apple has any meaningful relationship with the Western District of Wisconsin. Nokia is incorporated under the laws of Finland with its principal place of business in Finland. (Compl. ¶ 2.) Nokia's indirect U.S. subsidiary, Nokia Inc., is a Delaware corporation with its principal place of business in Texas. Apple Inc. is a California corporation with its principal place of business in Cupertino, California; it has no corporate offices or research facilities in Wisconsin. (Bentley Decl. at ¶¶ 2, 6, 7.)

Given the parties' lack of any meaningful contacts with Wisconsin, it is unlikely that any material fact witnesses or documents will be located in Wisconsin. *Id.* at ¶ 7. Apple and Nokia do not conduct relevant operations or maintain relevant facilities in Wisconsin.[6] *Id.* at ¶¶ 6, 7. Neither do any relevant third parties.

[5] HTC has not answered Apple's complaints in the 166, 167, and 544 Cases.

[6] Apple's principal place of business is in California; both Apple and Nokia maintain relevant facilities in California; and several third parties that manufacture components for relevant Apple products are based in California or maintain offices in California. A number of witnesses and documents may be overseas. Nokia is based in Finland, and the available evidence suggests that all of the named inventors for the asserted Nokia patents reside either in Finland or Great Britain. Moreover, a number of potentially relevant third-party manufacturers are based overseas, including Foxconn Electronics Inc. (based in Taiwan).

2. Nokia Made The Choice To Litigate This Business Dispute Between Nokia And Apple In Delaware

Nokia cannot deny that the District of Delaware is a convenient forum. Indeed, Nokia made the choice to litigate this ongoing business dispute between Nokia and Apple in the District of Delaware—where its U.S. subsidiary, Nokia Inc., is incorporated—by filing its first two lawsuits against Apple in that forum. (*See* Ex. 3, Delaware I Compl.; Ex. 5, Delaware II Compl.)

In fact, Nokia has chosen Delaware not only for its larger dispute with Apple, but also for comparable disputes with two other major telecommunications companies, Interdigital Communications Corporation and Interdigital Technology Corporation (No. 1:05-CV-00016) (Jan. 12, 2005), and Qualcomm, Inc. (No. 1:06-CV-00509) (Aug. 16, 2006). Nokia filed those actions, like this one, after failed licensing negotiations. Moreover, Apple has chosen Delaware for its three disputes with HTC.

3. In Contrast, Nokia Has Moved To Transfer The Only Case Filed Against It In This District Because Nokia Lacks "Any Connection To This District Or The State Of Wisconsin"

In sharp contrast with its decision to sue Apple in Wisconsin, in the only prior case (according to PACER) in which Nokia appeared in this district, Nokia moved to transfer based on the company's lack of contacts with Wisconsin and its interest in consolidating the case with related litigation. (*See* Ex. 14, Nokia's Mot. to Transfer at 9-10, *Qualcomm Inc. v. Nokia Corp. et al.*, No. 3:07-CV-0187 (W.D. Wis. May 24, 2007).) Nokia argued, in its motion, that Wisconsin was an inconvenient forum because neither of the parties had "any connection to this district or the state of Wisconsin." (Ex. 1, Nokia Reply In Support Of Motion to Transfer at 1 (emphasis in original).) It also emphasized the efficiencies of consolidating related cases involving the same general technology, *id.* at 7, the same types of efficiencies that could be accomplished by transferring this case to the District of Delaware.

IV. Argument

A. The Legal Standard For Motions To Transfer Under 28 U.S.C. § 1404(a)

The federal venue statute provides that, "[f]or the convenience of parties and witnesses, in the interest of justice, a district court may transfer any civil action to any other district or division where it might have been brought." 28 U.S.C. § 1404(a). In considering motions under § 1404(a), courts generally analyze the plaintiff's choice of forum as well as the statutory factors of (1) convenience to the parties; (2) convenience to the witnesses; and (3) the interests of justice. *See Uniroyal Engineered Prods., L.L.C. v. Omnova Solutions Inc.*, No. 08-CV-586-SLC, 2009 WL 736700, at *3 (W.D. Wis. Mar. 19, 2009). However, a plaintiff's choice of forum deserves no deference unless the plaintiff is "litigating in [its] home forum." *Id.*

The Federal Circuit, which maintains appellate jurisdiction over this case, has emphasized the importance of careful application of venue principles in a series of recent decisions. In *In re TS Tech USA Corp.*, 551 F.3d 1315 (Fed. Cir. 2008), for example, the court found that the district court had "clearly abused its discretion in denying transfer from a venue with no meaningful ties to the case." *Id.* at 1321 (emphasis added) (issuing writ of mandamus transferring venue); *see also In re Nintendo Co.*, 589 F.3d 1194, 1198 (Fed. Cir. 2009) (issuing writ of mandamus transferring venue) ("in a case featuring most witnesses and evidence closer to the transferee venue with few or no convenience factors favoring the venue chosen by the plaintiff, the trial court should grant a motion to transfer"); *In re Genentech, Inc.*, 566 F.3d 1338, 1348 (Fed. Cir. 2009) (issuing writ of mandamus transferring venue) ("There are no witnesses or parties within Texas. Moreover, there are no relevant documents there. … Thus, the convenience to the witnesses and parties, availability of compulsory attendance and access to evidence factors all weigh significantly in favor of transfer."); *In re Hoffmann-La Roche Inc.*, 587 F.3d 1333, 1336 (Fed. Cir. 2009) (issuing writ of mandamus transferring venue) ("As in

Volkswagen, *TS Tech*, and our most recent decision, *In re Genentech, Inc.*, 566 F.3d 1338 (Fed. Cir. 2009), there is a stark contrast in relevant, convenience, and fairness between the two venues.").

B. The Convenience Of The Parties, The Convenience Of The Witnesses, The Interests Of Justice, And Basic Common Sense All Weigh Strongly In Favor Of Transfer To Delaware

Nokia has already chosen the District of Delaware for this business dispute between Nokia and Apple. This case should be transferred to that district, where it can be consolidated with Nokia's two other recently-filed cases, which are also based on Apple's iPhone products, and Apple's two related cases against HTC. Given the obvious benefits of trying all the concurrent and related cases in the same district court—not to mention Nokia's previous insistence that it should not litigate in this Court because it has no ties to the Western District of Wisconsin—the District of Delaware is a clearly more convenient forum for the parties and the witnesses, and transfer would serve the interests of justice.

1. The Parties' Lack Of Any Meaningful Connection To Wisconsin Supports Transfer And Means That Nokia's Choice Of Forum Deserves No Weight

Nokia's choice of forum deserves no weight in the transfer analysis, because Nokia is not "litigating in [its] home forum," *Uniroyal*, 2009 WL 736700, at *3, and because "this case has no discernible connection to this district" or even "to this state." *Lineage Power Corp. v. Synqor, Inc.*, No. 08-CV-397-SLC, 2009 WL 90346, at *5 (W.D. Wis. Jan 13, 2009). As Nokia has admitted, neither Apple nor Nokia has "any connection to this district," (Ex. 1, Nokia Reply In Support Of Motion to Transfer at 1), and it is highly unlikely that any relevant fact witness resides or is employed in Wisconsin. (Bentley Decl. at ¶ 7.) This complete lack of connection to Wisconsin "militates toward transfer." *Lineage Power Corp.*, 2009 WL 90346, at *5; *see also, e.g.*, *U.S.O. Corp. v. Mizuho Holding Co.*, 547 F.3d 749, 752 (7th Cir. 2008) ("The more tenuous

a party's relation to the forum, the weaker the case for litigating there."); *cf. Chicago, Rock Island and Pac. R.R. Co. v. Igoe*, 220 F.2d 299, 304 (7th Cir. 1955) ("this factor has minimal value where none of the conduct complained of occurred in the forum selected by the plaintiff").

2. The District Of Delaware Is A Clearly More Convenient Forum For The Parties And The Witnesses

The District of Delaware is a clearly more convenient forum because the parties are already conducting related litigation in that district.[7] The parties and their principal lawyers will be litigating in the District of Delaware regardless of where this case proceeds.[8] The same applies to many of the witnesses in this case, because their testimony will be required in Delaware.

Thus, while neither the parties nor the witnesses have any significant ties to Wisconsin, both parties and many of the witnesses will be appearing in the District of Delaware regardless of where this case proceeds. Moreover, as discussed more fully below, "related litigation should be transferred to a forum where consolidation is feasible," *Coffey v. Van Dorn Iron Works*, 796 F.2d 217, 221 (7th Cir. 1986), because consolidation permits effective coordination and can eliminate the need for duplicative appearances of both the parties and the witnesses. For all of these reasons, the convenience of the parties and the witnesses strongly weighs in favor of transfer to the District of Delaware.

[7] The mere fact that Nokia chose to file in this district does not demonstrate that the Western District of Wisconsin is a convenient forum for Nokia. *See Piper Aircraft Co.*, 454 U.S. 235, 255-56 (1981) ("when the home forum has been chosen, it is reasonable to assume that this choice is convenient. When the plaintiff is foreign, however, this assumption is much less reasonable.").

[8] The lead law firms for Nokia and Apple are the same in all of the relevant cases.

3. The Interests Of Justice Require Transfer To The District Of Delaware

The interests of justice, which are often "determinative" in transfer motions, are primarily concerned with "the efficient administration of the court system." *Id.* The interests of justice "may be determinative . . . even if the convenience of the parties and witnesses might call for a different result." *Id.* at 220. In this case, the interests of justice weigh decisively in favor of transfer to the District of Delaware.

One of the most important factors in assessing the "interests of justice" is whether transfer would permit related litigation to proceed in the same district. *See Heller Fin., Inc. v. Midwhey Powder Co.*, 883 F.2d 1286, 1293 (7th Cir. 1989) ("[t]he 'interest[s] of justice' include such concerns as ensuring speedy trials, trying related litigation together, and having a judge who is familiar with the applicable law try the case."). Transfer is particularly appropriate where, as here, it would facilitate consolidation of related cases. *See Coffey*, 796 F.2d at 221 ("[R]elated litigation should be transferred to a forum where consolidation is feasible."). As the Supreme Court has emphasized, litigating the same issues in different courts inevitably "leads to the wastefulness of time, energy and money that [28 U.S.C.] § 1404(a) was designed to prevent." *Continental Grain Co.*, 364 U.S. at 26; *see also Lineage Power Corp.*, 2009 WL 90346, at *1 ("efficiency and common sense" favored transfer to a district where related cases were already pending); *D2L Ltd. v. Blackboard, Inc.*, 671 F. Supp. 2d 768, 783 (D. Md. 2009) (citing "economies" in conducting related cases in one forum); *First Health Group Corp. v. Allcare Health Mgmt. Sys., Inc.*, No. 01-CV-1790, 2001 WL 686777, at *2 (N.D. Ill. June 15, 2001) ("As a rule, cases should be transferred to the district where related actions are pending.").

It is well recognized, for example, that "consolidation . . . provides genuine opportunities measurably to streamline discovery," and "makes it possible to save some witnesses time and money by consolidating their depositions, affidavits and other evidentiary input." *Lineage*

Power Corp., 2009 WL 90346, at *6. Conducting the litigation here in separate courts would, for example, require repeated depositions from the same party and third-party witnesses in this case as well as the Delaware cases. *See, e.g., Encyc. Britannica, Inc. v. Magellan Navigation, Inc.*, 512 F. Supp. 2d 1169, 1177 (W.D. Wis. 2007) ("Since the same or related patents are at issue it is probable that defendants from all cases will depose the same persons and request the same documents and technical drawings from plaintiff.").

Litigating the parties' Delaware and Wisconsin claims in multiple actions also would considerably complicate the discovery, pretrial, and trial processes, and would almost certainly result in two separate courts considering the same procedural, evidentiary and substantive issues. For example, during the discovery phase of the case, the Delaware and Wisconsin cases are likely to involve the same disputes about matters such as protective orders, motions to compel and quash discovery, and third-party discovery. As the claims approach trial, the cases are likely to involve similar motions in limine, evidentiary disputes, and disputes over jury instructions. Litigating the larger dispute in multiple cases would require two courts to consider these same issues, resulting in an inefficient use of judicial resources, and a very real risk of conflicting rulings. *See, e.g., Seiko Epson Corp. v. Optoma Tech., Inc.*, No. 06-CV-06946-MJJ, 2007 WL 1793776, at *3 (N.D. Cal. Jun. 19, 2007) (ordering transfer where "parallel litigation in two different forums would create a serious risk of conflicting pretrial and trial schedules, not to mention inconsistent management of discovery limits and specialized patent litigation procedures"); *In re Ephedra Prods. Liab. Litig.*, 314 F. Supp. 2d 1373, 1375 (J.P.M.L. 2004) ("[c]entralization . . . is necessary in order to avoid . . . inconsistent or repetitive pretrial rulings, and conserve the resources of the parties, their counsel and the judiciary").

The benefits of consolidation are particularly important in complex patent cases, where consolidation "reduce[s] the need for duplicative time-consuming [technological] tutorials" necessary for both the Court and the jury to understand the issues in dispute. *Broadcom Corp. v. Agere Sys. Inc.*, No. 04-CV-066-C, 2004 WL 1176168, at *1 (W.D. Wis. May 20, 2004); *see also Lineage Power Corp.*, 2009 WL 90346, at *6 (describing efficiencies where there is a "concrete overlap between the technology"). Consolidation also reduces the risk of conflicting rulings on claim constructions for the same patents. *See D2L Ltd.*, 671 F. Supp. 2d at 784 (where patents have the same inventors, drawings and specifications, "transfer will promote judicial economy and avoid the possibility of inconsistent construction of the [patents-in-suit], which cover the same technology and have many overlapping claims."); *DataTreasury Corp. v. First Data Corp.*, 243 F. Supp. 2d 591, 595 (N.D. Tex. 2003) (transfer is especially appropriate "in patent cases, where the court is required to interpret the claims of the patent in-suit.").

Moreover, to the extent that either of the courts attempts to promote a negotiated settlement, negotiations in one case will be inextricably intertwined with the discussions in the other case. Allowing the related claims to proceed in separate actions could reduce the settlement pressure on the parties and complicate settlement negotiations. *See Digeo, Inc. v. Gemstar-TV Guide Int'l, Inc.*, No. 06-CV-1417RSM, 2007 WL 295539, at *5 (W.D. Wash. Jan. 29, 2007) (citing increased likelihood of settlement through consolidation of related claims); *Solaia v. Rockwell*, No. 03-CV-566, 2003 WL 22057092, at *3 (N.D. Ill. Sept. 2, 2003) ("[I]f these cases proceed in different districts, there is little if any possibility of consolidating them for discovery, settlement negotiations or trial."); *Amazon.com v. Cendant Corp.*, 404 F. Supp. 2d 1256, 1262 (W.D. Wash. 2005) ("While consolidation of the two cases is a matter for the Delaware court to decide, the feasibility of such consolidation is a factor that this Court may

consider in deciding whether to allow a transfer. Even if they are not consolidated, transfer of this case will allow a global settlement of the related cases.").

These same benefits will accrue even if the cases are not fully consolidated, particularly here, where all of the litigation is in its early stages. For example, even without full consolidation, litigating all cases in the same forum will facilitate coordination of the practical aspects of discovery, thereby reducing duplicative efforts and preserving the resources of the courts and parties and facilitating the resolution of any discovery disputes. Similarly, the parties and the courts will benefit from litigating disputes before a court that is familiar with the parties, their products, and the essential technology. *See, e.g.*, *Amazon.com*, 404 F. Supp. 2d at 1261 (cases involving different software patents were "similar enough that they should be considered by the same court in order to conserve judicial resources and prevent inconsistent rulings"); *Broadcom Corp. v. Microtune, Inc.*, No. 03-CV-0676-S, 2004 WL 503942, at *4 (W.D. Wis. Mar. 9, 2004) ("Although the Eastern District has not construed the claims of the '742 patent, it is familiar with the general silicon-based tuner technology and specific accused devices at issue in the current dispute. . . . Consequently, the Eastern District is in the best position to manage the parties efficiently and achieve a prompt resolution of this dispute."); *Abbott Labs. v. Selfcare, Inc.*, No. 98-CV-7102, 1999 WL 162805, at *2 (N.D. Ill. Mar. 15, 1999) ("The two actions, even though directed to different patents, involve the same parties and substantially similar technology. They also involve similar complex factual and legal questions that will require the expenditure of considerable time and effort. Requiring two courts to devote limited resources educating themselves about the same underlying technology would undermine values of judicial economy.").

V. Conclusion

Nokia has already filed two related cases in the District of Delaware; those cases are legally, factually, and procedurally intertwined with this case; all three cases are in the early stages of the litigation; and all three cases arise out of the same business dispute based on the same technology and same products. Moreover, Apple has filed three related cases against HTC in the District of Delaware; Apple has moved to consolidate these cases with the two Nokia cases; and three of the patents that Apple has asserted in this case have also been asserted against HTC in Delaware. In this context, the District of Delaware is a clearly more convenient forum for the parties and the witnesses, and the interests of justice plainly compel transfer. For all of these reasons, Apple respectfully requests that the Court transfer this case to the United States District Court of Delaware.

GODFREY & KAHN, S.C.

By: *s/James D. Peterson*
James Donald Peterson (# 1022819)
One East Main Street, Suite 500
P.O. Box 2719
Madison, WI 53701-2719
Tel: (608) 257-3911
Fax: (608) 257-0609
jpeterson@gklaw.com

Attorneys for Defendant and Counterclaim-Plaintiff Apple Inc.

OF COUNSEL:

WILMER CUTLER PICKERING
HALE AND DORR LLP

William F. Lee
Cynthia D. Vreeland
Dominica E. Massa
60 State Street, Boston, MA 02109
Tel. (617) 526-6000

Keith Slenkovich
1117 California Avenue
Palo Alto, CA 94304
Tel. (650) 858-6000

Dated: June 29, 2010

5137257_2

EXHIBIT 4

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Matheny et al.
U.S. Patent No.:	5,315,703
Issued:	May 24, 1994
Group Art Unit:	2317
Serial No:	07/996,782
Examiner:	D. Shaw
Filed:	Dec. 23, 1992
For:	OBJECT-ORIENTED NOTIFICATION FRAMEWORK SYSTEM
Attorney Docket No.	0919/01032

April 28, 2010

Mail Stop *Ex Parte* Reexamination
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

REQUEST FOR REEXAMINATION

Reexamination of United States Patent 5,315,703 (hereinafter, "the '703 patent"), which issued May 24, 1994, to Matheny, White, Anderson, and Schaeffer is requested under 35 U.S.C. §§ 302-307, and under 37 C.F.R. § 1.510. This patent is still in force.[1] A copy of the patent in accordance with 37 C.F.R. § 1.510(b)(4) is submitted herewith as Exhibit A.

I. Claims for which Reexamination is Requested

The '703 patent is for a change notification system in an object-oriented environment. The patent discloses a framework where a connection object stores connection registration information in the memory of a computer system. When a notification event is detected, the

[1] A lawsuit for the alleged infringement of the '703 patent has been filed in the U.S. District Court of Delaware, *Nokia Corp. v. Apple Inc.*, Civil Action No. 1:09-cv-00791-GMS. That litigation is in its early stages and no discovery regarding the '703 patent have taken place. If the litigation proceeds, Third Party Requestor expects there will be a challenge to the validity of the '703 patent therein.

system selectively notifies objects based on the connection registration information in the connection object. Reexamination is requested of all Claims 1-14 of the '703 patent.

II. Statement of Substantial New Questions of Patentability

A. The Subject Matter of Claims 1-14

Claims 1-14 recite:

1. *An object-oriented notification framework system, comprising:*

   means for connecting a plurality of objects to a notification source;

   memory means for storing connection information for the plurality of objects in a connection object of an object-oriented operating system;

   means for registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;

   means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system; and

   means for the at least one of the plurality of objects to receive the notification and take action based on the notification.

2. *A system as recited in claim 1, including a processor means for notifying a plurality of objects.*

3. *A system as recited in claim 1, including processor means for changing a color of an object as an action based on notification.*

4. *A system as recited in claim 1, including processor means for highlighting an object as an action based on the notification.*

5. *A system as recited in claim 1, including processor means for reverse videoing an object as an action based on the notification.*
6. *A system as recited in claim 1, including processor means for removing an object as an action based on the notification.*
7. *A system as recited in claim 1, including processor means for opening a window associated with an object as an action based on the notification.*
8. *A method for implementing an object-oriented notification framework system, comprising the steps of:*

   connecting a plurality of objects to a notification source;

   storing connection information for the plurality of objects in a connection object of an object-oriented operating system;

   registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;

   selectively dispatching notification to at least one of the plurality of objects based on the connection registration information stored in the connection object of the object-oriented operating system; and

   receiving the notification by the at least one of the plurality of objects and taking action based on the notification.
9. *A method as recited in claim 8, including the step of notifying a plurality of objects.*
10. *A method as recited in claim 8, including the step of changing the color of an object as an action based on the notification.*
11. *A method as recited in claim 8, including the step of highlighting an object as an action based on the notification.*

12. A method as recited in claim 8, including the step of reverse videoing an object as an action based on the notification.

13. A method as recited in claim 8, including the step of removing an object as an action based on the notification.

14. A method as recited in claim 8, including the step of opening a window associated with an object as an action based on the notification.

In reexamination, as with any proceeding before the U.S. Patent and Trademark Office ("USPTO"), the terms and phrases of a claim are given their broadest reasonable construction. *E.g., In re American Academy of Science Tech Center*, 367 F.3d 1359, 70 USPQ2d 1827, 1830 (Fed. Cir. 2004) ("During examination, 'claims ... are to be given their broadest reasonable interpretation . . .." (*quoting In re Bond*, 910 F.2d 831, 833, 15 USPQ2d 1566 (Fed. Cir. 1990))).

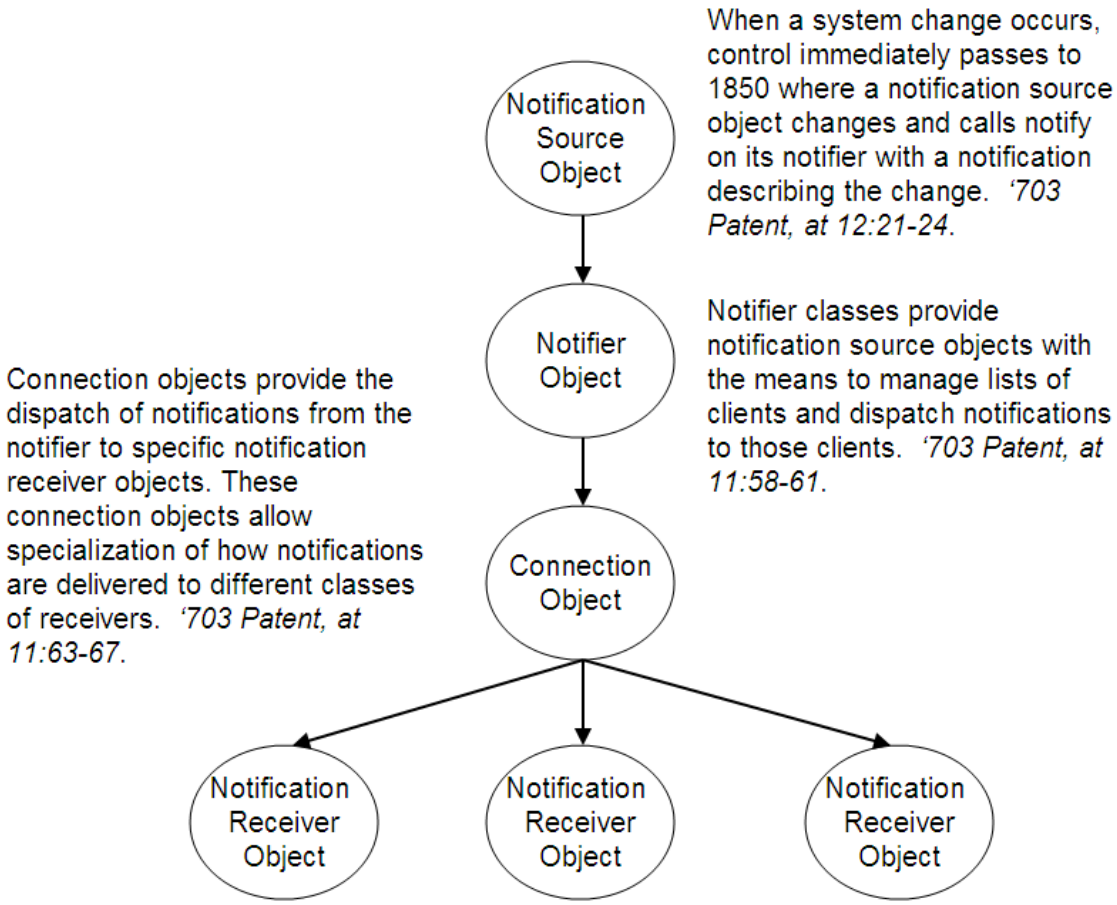
The '703 patent claims relate to an object-oriented change notification system. The claims describe an apparatus and means for connecting a plurality of objects to a notification source through a connection object. The connection object contains connection information specifying the registered interests specific to each of the plurality of objects. Upon a change to the source object, the connection object dispatches notification to one or more receiver objects in the plurality of objects based on the registration information. The structure of the embodiment of the invention is disclosed in the '703 specification at columns 11:29 – 12:39.

A chart relating the '703 claim elements (for exemplary claim 1) to the '703 specification follows.

Exemplary Claim 1	Language from '703 specification
1. An object-oriented notification framework system, comprising:	
(a) means for connecting a plurality of objects to a	"The client object asks the connection object to connect to the notification source(s) for notifications specified by the interests

notification source;	in the connection…" *'703 Patent, at 12:14-16.*
(b) memory means for storing connection information for the plurality of objects in a connection object of an object-oriented operating system;	"The change notification includes a memory for storing connection information including notification routing information and connection registration information. The connection registration information is stored in a connection object of the object-oriented notification system and the notification system updates the connection object with registration information indicative of enablement or disablement of notification." *'703 Patent, at Abstract.*
(c) means for registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;	"[F]or each interest in connection, the connection is registered as interested in the notification with the notifier in the interest." *'703 Patent, at 12:17-19.*
(d) means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system; and	"When a system change occurs … a notification source object changes and calls notify on its notifier with a notification describing the change." *'703 Patent, at 12:21-24.* "For each connection registered with the notifier as interested in the notification … the connection is asked to dispatch the notification." *'703 Patent, at 12:25-27* "Connection objects provide the dispatch of notifications from the notifier to specific notification receiver objects." *'703 Patent, at 11:63-67.*
(e) means for the [sic] at least one of the plurality of objects to receive the notification and take action based on the notification.	"[T]he connection dispatches the notification to the appropriate method of the notification receiver. Finally, at function block 1880, the notification receiver takes the appropriate action for the notification…" *'703 Patent, at 12:28-32.*

In describing the invention, independent claims 1 and 8 refer to three separate types of objects in the system: a "plurality of objects" (referring to one or more "notification receiver objects"), "notification source objects", and a "connection object". The specification discloses an additional "notifier object." A diagram depicting the structure of the objects used in the invention follows:



B. Newly Cited Prior Art

The '703 patent was filed December 23, 1992, and claims no earlier priority filing date. Therefore, the "Critical Date" for prior art relevant to the claims of the '703 patent, under 35 U.S.C. § 102(b) is December 23, 1991.

The requester respectfully submits that the prior art, under §§ 102(b), 102(e), and/or § 103 taught or suggested the subject matter of the claims of the '703 patent. More particularly, the requester submits that:

- E. Cohen, D. Soni, R. Gluecker, W. Hasling, R. Schwanke, M. Wagner, "Version Management in Gypsy" (ACM 1988) pages 201, 210-211 ("the Cohen article") (Exhibit B);

anticipated the subject matter of the claims of the '703 patent. Furthermore, the Cohen article alone, and in combination with

- U.S. Patent No. 5,204,947 to Bernstein et al. ("the Bernstein patent") (Exhibit C)

rendered the subject matter of the claims of the '703 patent obvious to one of ordinary skill in the relevant art.

Furthermore, the requester notes that the Cohen article, and the Bernstein patent were not listed on the face of the '703 patent. Consequently, the Cohen article and the Bernstein patent are newly applied and unquestionably raise new questions of patentability.

C. Basis for Substantial New Questions of Patentability

Claims 1-14 of the '703 patent do not patentably distinguish over the above-noted newly cited references, either alone or in combination. In summary, the Cohen article discloses a programming support environment built on top of an object-oriented operating system. The programming environment includes components for object-oriented event handling, where an Event Manager provides support for monitoring changes to objects and triggering actions (including dispatching notifications) when a change event occurs. In the programming environment disclosed in Cohen, users express interest in a change event by subscribing to a notification source in a target object. An Event Manager provides the support for linking subscribing users to target objects. When a change event occurs, the target object's type manager signals the Event Manager. The Event Manager then triggers an action for the specific subscribed users who expressed interest in that target object. Before triggering the action, the Event Manager checks the authority list of the targeted object to ensure the subscriber still has

rights to access it. For the action, the subscriber might choose to simply receive notification or execute a different program when the event occurs. (Exhibit B, pp. 201, 210-211).

The Cohen article describes various actions that a subscribing individual (or a program associated with the subscribing individual) can take in response to receiving notification of changes to a particular target object (e.g., changes to a version of a particular document). For example, Cohen teaches that a user can unlock or release objects in response to the occurrence of changes to particular target objects that the user is monitoring. (Exhibit B, pp. 211-212).

In obtaining the '703 patent, the patent owner stated:

> The crux of the invention is an object-oriented system with the capability of selectively enabling or disabling notification to objects based on information stored in a connection object. (Amendment, Sept. 23, 1993, Serial No. 07/996,782).

The Cohen article meets the alleged deficiencies in the prior art of record in the original prosecution by providing an object-oriented operating system with the capability of selectively enabling or disabling notification to objects through the use of an Event Manager. Specific types of events can be monitored via different type managers, and users express interests in particular events through subscriptions to target objects. Selective dispatch occurs as the Event Manager triggers actions only for the specific subscribed users who expressed interest in the notification. Subscribers can selectively cancel particular events as well, thus disabling notification. Subscribing individuals, or programs associated with the subscribing individuals, can then take action (e.g., automatically) in response to receiving a notification from the Event Manager.

The Bernstein patent discloses a method and system for providing a consistent graphical user interface completely managed by underlying linking services. Individual applications are displayed in separate application windows on a workspace and consistently managed. As part of that consistent user interface, the Bernstein patent discloses the simple actions of highlighting an

object, changing the color of an object, and inverting the colors of an object. (Exhibit C, Col 2:33-51).

Thus, the Cohen article discloses and anticipates each element of independent claims 1, 2, 6, 8, 9 and 13 of the‘703 patent. The Cohen article also discloses or would have rendered obvious by itself each element of all claims 1-14 of the ‘703 patent. To the extent the patent owner may argue that the Cohen article might not teach certain claim elements of the ‘703 patent, those elements would have been obvious in view of the combination of the Cohen article and the Bernstein patent.

One skilled in the art would have found a clear motivation to combine the Cohen article with the teachings of the Bernstein patent. For example, one skilled in the art would have been motivated to highlight, change the color of, or reverse video a particular object, or open a new display window, on a computer display screen associated with a subscribing individual. This would serve to alert the individual that they have received notification of changes to an object that the individual is monitoring.

Because the Cohen article and the Bernstein patent were not previously considered and are not cumulative of any reference previously considered, combination of these references necessarily raises a new, and not cumulative, question of patentability. Consequently, Reexamination of claims 1-14 of the ‘703 patent must be ordered and all the claims rejected.

D. Application of Prior Art References to Claims 1-14

1. Content of the Prior Art

(a) The Cohen Article

The Cohen article discloses a programming support environment built on an object-oriented operating system. Integrated in the Cohen system is an Event Manager component that

provides support for monitoring changes to objects and triggers actions when such a change occurs:

> The Gypsy Programming Support Environment provides support for a team of developers to produce and maintain systems built from multiple components. The foundation of Gypsy is the Version Manager, which is used to store, organize, and selectively retrieve versions of objects. Integrated with version management are components for workspace, configuration, and event management:
> . . .
>
> - The Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. In particular, it can be used to support tools (such as Marvel [Kaiser87]) that selectively pre-build products as new versions of components are released.
>
> . . .
> More than any other part of the system, the Version Manager is tied to the operating system. Gypsy is built on top of an object-oriented operating system which provides mechanisms that support especially clean integration. (p. 201).

The Cohen article teaches using the Event Manager for object-oriented event handling. Since the Cohen article clearly discloses a change notification system in an object-oriented environment, one skilled in the art would understand that the Event Manager is implemented as an object data structure. In the Gypsy system, the user expresses an interest in a change event by making a subscription to a target object. The Event Manager (connection object) then links the subscribed user to the "type manager" (notifier) of the target object (notification source object). The type manager monitors the target object for the change event. When an event is detected, the type manager signals the Event Manager:

> 9. Event Management
> The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers.
>
> The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event

> occurs.
>
> An event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs. A recurring subscription remains in effect until it is explicitly canceled.
>
> 9.1. Object-Oriented Event Handling
> The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object, and for monitoring the target for the occurrence of the event. A type manager agrees to monitor events by supplying an EVENT attribute with two methods: a "subscription" method which takes the target object and the condition, and which returns a subscription, and a "cancel" method which takes the target object and the subscription, and cancels it.
>
> On a subscription, the Event Manager passes the *condition* on to the type manager without any interpretation; each type manager determines the sort of events it will monitor and interprets the condition accordingly. When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. The action is executed under the authority of the subscribing user. (p. 210-211).

Thus, when an event is detected, the Event Manager triggers an action only for the subscriber that has expressed an interest in the particular event. Also, when an event is detected, the Event Manager object first checks an authority list of the target object to ensure that the subscriber has the right to access it:

> Before triggering the action, the Event Manager first checks the authority list of the targetted [sic] object to ensure that the subscriber still has the right to access it. If not, the subscription is canceled and no action is triggered. (p. 211).

In this manner, the Event Manager only dispatches notification of change events to subscribed users that are interested in the particular event and who have authority to access the target object. The subscribed users (e.g., subscribing individuals or computer programs associated with those individuals) may then take various actions based on the dispatched notifications.

Further, the Cohen reference teaches object-oriented event handling for a variety of different events and conditions, including events for monitoring object version control and management:

> 9.2 Version Management Events
>
> The Version Manager supports subscriptions to events for version group objects. It supports a variety of conditions, which effectively allow monitoring of events such as:
>
> - When a specified operation (e.g. release, destroy, change property) is taken on any version satisfying a specified predicate.
> - When a specified operation (e.g. lock, unlock, release version) is taken on a specified branch.
> - When as a result of some operation on a version group, a specified version no longer satisfies a specified predicate, or a new version is selected by a specified selector. (p. 211).

(b) The Bernstein Patent

The Bernstein patent discloses a system and method to implement a consistent graphical user interface through the use of underlying hypermedia services. In the implementation of this user interface, the Bernstein patent describes the use of "Link markers" as a visual indication to the end user that a link object exists. (Exhibit C, col. 1:15-25, col. 2:21-50). The underlying hypermedia services are used to portray and update link markers with varying visual styles, including highlighting, changing the color, and reverse videoing, the corresponding link markers. (Exhibit C, col 2:33-50). The Bernstein patent also discusses displaying application data on multiple windows of a computer system. (Exhibit C, col. 33:38 – col. 34:5). The Bernstein patent discloses such a system in the context of an object-oriented programming environment. (Exhibit C: col. 3:6-29).

2. Grounds for Rejection of the Claims

(a) **First Ground for Rejection:** Claims 1, 2, 6, 8, 9, and 13 were anticipated by the Cohen article under 35 U.S.C. § 102(b).

Claim	Prior Art
1. An object-oriented notification framework system, comprising:	The Cohen reference describes an object-oriented notification system. In particular, the Cohen reference describes a programming support environment that is integrated on an object-oriented operating system. This programming environment provides components for monitoring changes to objects. This paper describes the Version Manager of the Gypsy programming support environment, and its integration with the object-oriented extension of Unix on which it is built. *Abstract.* The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. *Pg. 210.*
1[a] means for connecting a plurality of objects to a notification source; The '703 patent specification describes implementing this function : "The client object asks the connection object to connect to the notification source(s) for notifications specified by the interests in the connection…" *'703 Patent, at 12:14-16*.	This element is described in the Cohen reference. The Gypsy Environment provided a system to store, organize, and selectively retrieve multiple versions of objects. Integrated within the Gypsy Environment was an event management system, where subscribers (a plurality of objects, also called notification receiver objects in the '703 patent) were connected to a target object (notification source object in the 703 patent). Connection occurred through an Event Manager, which is the connection object as described in the 703 patent. The Gypsy Programming Support Environment provides support for a team of developers to produce and maintain systems built from multiple components, The foundation of Gypsy is the Version Manager, which is used to store, organize, and selectively retrieve versions of objects. Integrated with version management are components for workspace, configuration, and event management. *Pg. 201(Emphasis added).*. **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. *Pg. 210(Emphasis added).* The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210, (Emphasis added).* **9.1. Object-Oriented Event Handling** The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object… *Pg. 210.* As disclosed in the Cohen reference, subscribing users ask the Event Manager to connect to the correct target object. Thus, the Cohen reference explicitly discloses this claim element, even if it is limited to the structure disclosed in the specification of the '703 patent.
1[b] memory means for	This element is disclosed in the Cohen reference. The Gypsy Environment contains a system for

storing connection information for the plurality of objects in a connection object of an object-oriented operating system; The '703 patent specification describes implementing this function : "The change notification includes a memory for storing connection information including notification routing information and connection registration information. The connection registration information is stored in a connection object of the object-oriented notification system and the notification system updates the connection object with registration information indicative of enablement or disablement of notification." *'703 Patent, at Abstract.*	object-oriented event handling, where an Event Manager object links subscribing plurality of objects and target objects via an Event Manager. The Event Manager (the connection object) stores connection information associated with the subscriptions: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210 (Emphasis added).* **9.1. Object-Oriented Event Handling** The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object, and for monitoring the target for the occurrence of the event…. When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. *Pgs. 210-211 (Emphasis added).*. In order for the Event Manager to trigger the appropriate action associated with the subscription, including the action of notifying subscribers, the Event Manager must necessarily store connection information for routing the particular events to the plurality of subscribers. The Event Manager also stores registration information indicative of enablement or disablement of notification: An event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs. A recurring subscription remains in effect until it is explicitly canceled. *Pg. 210*. A recurring subscription is indicative of "enablement of notification," while the act of canceling a recurring subscription is indicative of "disablement of notification." Furthermore, the Event Manager contains an authority list of the target object informing it whether or not the subscriber has the right to access it: Before triggering the action, the Event Manager first checks the authority list of the targetted [sic] object to ensure that the subscriber still has the right to access it. If not, the subscription is canceled and no action is triggered. The user is not notified of the cancelation… Pg. 211. Whether or not the subscriber has the right to access the target object is further evidence of enablement or disablement of notification. A person with ordinary skill in the art would understand that an Event Manager object for storing connection information must necessarily be implemented via some type of memory means in a computer system. Thus, the Cohen reference explicitly discloses this claim element, even if it is limited to the structure disclosed in the specification of the '703 patent.
1[c] means for registering connection information, including registration information indicative of a	This element is disclosed in the Cohen reference. The Gypsy Environment discloses a system for object-oriented event handling, where an Event Manager object links subscribing users and target objects through subscriptions. Each subscription includes an action, which corresponds to registration information indicative of a notification status as described in the '703 patent:

notification status, in the connection object of the object-oriented operating system; The '703 patent specification describes implementing this function : "[F]or each interest in connection, the connection is registered as interested in the notification with the notifier in the interest." *'703 Patent, at 12:17-19*.	**9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. An event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs. A recurring subscription remains in effect until it is explicitly canceled. *Pg. 210 (Emphasis added).* As shown in the Cohen reference, subscriptions contain "actions", which consist of information indicating whether or not a subscriber is to receive event notification. It also contains information indicating whether the event is a one-time event or a recurring event. The information noting whether or not a subscriber is receiving notification and whether that notification is recurring corresponds to "notification status" from the '703 patent. Thus, the Cohen reference discloses the step of registering connection information, including registration information indicative of notification status, in the connection object of the object-oriented operating system. The Cohen reference explicitly discloses this claim element, even if it is limited to the structure disclosed in the specification of the '703 patent. For each subscription in the system, the subscriber is registered as interested in the notification with the type manager of the target object. The type manager is the "notifier object" as described in the '703 patent: **9.1. Object-Oriented Event Handling** The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object, and for monitoring the target for the occurrence of the event. A type manager agrees to monitor events by supplying an EVENT attribute with two methods: a "subscription" method which takes the target object and the condition, and which returns a subscription, and a "cancel" method which takes the target object and the subscription, and cancels it. *Pg. 210 (Emphasis added).*
1[d] means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system; and The '703 patent specification describes implementing this function : "When a system change occurs … a notification source object changes and calls notify on its notifier with a notification describing	This element is disclosed in the Cohen reference. The Gypsy Environment contains an object-oriented event handling system for monitoring changes to objects. When a target object (corresponding to a notification source object in the '703 patent) changes, change notification is selectively dispatched to the subscribed users interested in the event. As used in the Cohen reference, the term "user" could be reasonably understood to mean either: (1) a particular individual (a subscribing individual); or (2) an object (e.g., a program) associated with that particular individual. One skilled in the relevant field would understand that dispatching a notification to a particular individual would typically involve electronically sending the notification to a computing device associated with the individual, and that the user's computing device would include a program (e.g., an object) for presenting the notification to the individual in an appropriate manner. The Cohen reference discloses how subscribing users express interest in specific events. When the target object changes, the Event Manager triggers the action (change notification) only for the specific subscribed users who expressed interest in that target object. Thus the Event Manager selectively dispatches the change events: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and

the change." *'703 Patent, at 12:21-24*. "For each connection registered with the notifier as interested in the notification … the connection is asked to dispatch the notification." *'703 Patent, at 12:25-27* "Connection objects provide the dispatch of notifications from the notifier to specific notification receiver objects." *'703 Patent, at 11:63-67*.	supports the inclusion of new types of events by individual type managers. Pg. 210 (*Emphasis added).* The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210(Emphasis added).* **9.1. Object-Oriented Event Handling** … On a subscription, the Event Manager passes the condition on to the type manager without any interpretation; each type manager determines the sort of events it will monitor and interprets the condition accordingly. When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. The action is executed under the authority of the subscribing user. *Pg. 211 (Emphasis added).* The Cohen reference discloses this claim element even if it is limited to the structure disclosed in the '703 patent specification. As in the '703 patent, when a system change occurs, a target object changes. The target object's type manager (the notifier object from the '703 patent) signals the Event Manager (the connection object). The Event Manager then provides the dispatch of notifications from the type manager to the specific subscribed users (notification receiver objects): The Event Manager posts a subscription to the target objects' type manager which is responsible for associating it with the target object, and for monitoring the target for the occurrence of the event. A type manager agrees to monitor events… *Pg 210 (Emphasis added*). When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. *Pg. 211 (Emphasis added).* The Cohen reference also describes a second disclosed means for selectively dispatching change events. The Event Manager triggers the action (sending notification to the subscribers) only if the corresponding subscribers still have the right to access the targeted object. This implementation of change notification thus allows the Event Manager to selectively dispatch change events: Before triggering the action, the Event Manager first checks the authority list of the targeted object to ensure that the subscriber still has the right to access it. If not, the subscription is canceled and no action is triggered. … Type managers which perform additional access control based on internal state (such as the Version Manager) will need to retain the identity of the subscriber internally, and when an event occurs, check internally whether the subscriber still has the right to monitor it. *Pg. 211*.
1[e] means for at least one of the plurality of objects to receive the notification and take action based on the notification. The '703 patent specification describes implementing this function : "[T]he connection dispatches the	This element is disclosed in the Cohen reference. The Gypsy Environment allowed for subscribed users to receive notification about changes to a target object and take actions based on the notification. In Gypsy, subscribing users (corresponding to the plurality of objects in the '703 patent) express an interest in specific events by making a subscription. At least one of the subscribing users that expressed an interest in the event can receive a notification of the event: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive

notification to the appropriate method of the notification receiver. Finally, at function block 1880, the notification receiver takes the appropriate action for the notification…"	notification, or to supply a program which will be executed when the event occurs. *Pg. 210(Emphasis added).* As noted above, as used in the Cohen reference, the term "user" could be reasonably understood to mean either: (1) a particular individual (e.g., a subscribing individual); or (2) an object (e.g., a program) associated with that particular individual. One skilled in the relevant field would understand that, within the context of a computer system such as the one described in Cohen, dispatching a notification to a particular individual would typically involve electronically sending the notification to a computing device associated with the individual, and that the individual's computing device would include a program (e.g., an object) for receiving the notification and taking action in response to receiving the notification (e.g., presenting the notification to the individual in an appropriate manner). In addition to receiving the notification, the subscribing individual (or, as would be understood by one skilled in the field, a program associated with the subscribing individual) can take various actions based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computing device that would serve to alert the subscribing individual that they have received notification of changes to an object that the individual is monitoring. The Cohen reference also describes various other actions that a subscribing individual (or a program associated with the subscribing individual) can take while monitoring changes to particular objects (e.g., versions of particular documents): We expect that a user will most often monitor the unlocking of a branch (so that it can lock it), and the release of a new version (so that it can be used in building a configuration). *Pg 211 (Emphasis added).* In a further example discussed below, Cohen teaches that a subscribing individual (or, as would be understood by one skilled in the field, a program associated with the subscribing individual) may automatically produce a new version of a document in response to the release of a new version of one of the document's chapters. Reading the Cohen reference as a whole, one skilled in the field would understand that this type of action would be taken in response to the subscribing individual receiving notification that the object has changed (e.g., that a new version of the document has been released). When objects are hierarchically organized, a user may want to version control both the top level object as well as the sub-objects. For example, if a document is made up of a collections of chapters, it may be useful to version control each of the chapters separately, and also maintain versions of the document as a whole. One method of coping with hierarchical versions is to percolate changes up the hierarchy; releasing a new version at one level automatically causes a release of a new version at each higher level [Zdonik86]. For example, releasing a new version of a chapter would automatically cause a new version of the entire document to be released. Hierarchical version management in Gypsy can be provided, more cleanly, and with finer control, through the use of configurations. If chapters are individually version controlled, then a document can be represented as a (version-controlled) configuration with each chapter as a component. A user can explicitly produce a new version of the document by making a new version of the configuration and binding its components. *Pg 212 (Emphasis added).*
2. A system as recited in claim 1, including processor means for notifying a plurality of	This element is disclosed in the Cohen article. The Gypsy Environment describes an Event Manager that provides support for monitoring changes to objects and notifying a plurality of objects when an event occurs:

objects.	The Gypsy Event Manager provides support for monitoring changes to objects, and triggering actions when an event occurs…. For the action, the subscriber may choose to either receive notification… *Pg. 210 (Emphasis added).* Thus, in the Gypsy Environment, the Event Manager can trigger a plurality of actions when an event occurs, each action corresponding to notification to a subscriber. The Cohen reference also discloses "Gypsy is built on top of an object-oriented operating system which provides mechanisms that support especially clean integration." *Pg. 201 (Emphasis added).* A person with ordinary skill in the art would understand that notification to a plurality of objects in an object-oriented operating system must necessarily be implemented through a processor means in a computer system.
6. A system as recited in claim 1, including processor means for removing an object as an action based on the notification.	This element is disclosed in the Cohen article. The Cohen article teaches triggering an action associated with a subscription upon the occurrence of a change event. "The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs." *(Pg. 210).* The article broadly defines action: "For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs." *Pg. 210 (Emphasis added).* The Cohen article further teaches "[a]n event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs." *Pg. 210 (Emphasis added).* In describing a one-time event subscription, the Cohen article effectively discloses the removal of an object as an action based on the notification. This is because in Gypsy, after a one-time event subscription is completed, the Event Manager triggers an action and the subscription is canceled, removing the object from future notifications.
8. A method for implementing an object-oriented notification framework system, comprising the steps of:	The Cohen reference describes an object-oriented notification system. In particular, the Cohen reference describes a programming support environment that is integrated on an object-oriented operating system. This programming environment provides components for monitoring changes to objects. This paper describes the Version Manager of the Gypsy programming support environment, and its integration with the object-oriented extension of Unix on which it is built. *Abstract.* The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. *Pg. 210.*
8[a] connecting a plurality of objects to a notification source;	This element is described in the Cohen reference. The Gypsy Environment provided a system to store, organize, and selectively retrieve multiple versions of objects. Integrated within the Gypsy Environment was an event management system, where subscribers (a plurality of objects, also called notification receiver objects in the '703 patent) were connected to a target object (notification source object in the 703 patent). Connection occurred through an Event Manager, which is the connection object as described in the 703 patent. The Gypsy Programming Support Environment provides support for a team of developers to produce and maintain systems built from multiple components. The foundation of Gypsy is the Version Manager, which is used to store, organize, and selectively retrieve versions of objects. Integrated with version management are components for workspace, configuration, and event management. *Pg. 201(Emphasis added).*. **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and

	supports the inclusion of new types of events by individual type managers. *Pg. 210(Emphasis added).* The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210, (Emphasis added).* **9.1. Object-Oriented Event Handling** The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object… *Pg. 210.*
8[b] storing connection information for the plurality of objects in a connection object of an object-oriented operating system;	This element is disclosed in the Cohen reference. The Gypsy Environment contains a system for object-oriented event handling, where an Event Manager object links subscribing plurality of objects and target objects via an Event Manager. The Event Manager (the connection object) stores connection information associated with the subscriptions: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210 (Emphasis added).* **9.1. Object-Oriented Event Handling** The Event Manager posts a subscription to the target object's type manager which is responsible for associating it with the target object, and for monitoring the target for the occurrence of the event…. When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. *Pgs. 210-211 (Emphasis added).*. In order for the Event Manager to trigger the appropriate action associated with the subscription, including the action of notifying subscribers, the Event Manager must necessarily store connection information routing the particular events to the plurality of subscribers.
8[c] registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;	This element is disclosed in the Cohen reference. The Gypsy Environment discloses a system for object-oriented event handling, where an Event Manager object links subscribing users and target objects through subscriptions. Each subscription includes an action, which corresponds to registration information indicative of a notification status as described in the '703 patent: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs.

	An event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs. A recurring subscription remains in effect until it is explicitly canceled. *Pg. 210 (Emphasis added).* As shown in the Cohen reference, subscriptions contain "actions", which consist of information indicating whether or not a subscriber is to receive event notification. It also contains information indicating whether the event is a one-time event or a recurring event. The information noting whether or not a subscriber is receiving notification and whether that notification is recurring corresponds to "notification status" from the '703 patent. Thus, the Cohen reference discloses the step of registering connection information, including registration information indicative of notification status, in the connection object of the object-oriented operating system. Thus, the Gypsy event handling environment includes a means for registering connection information in a connection object.
8[d] selectively dispatching notification to at least one of the plurality of objects based on the connection registration information stored in the connection object of the object-oriented operating system; and	This element is disclosed in the Cohen reference. The Gypsy Environment contains an object-oriented event handling system for monitoring changes to objects. When a target object (corresponding to a notification source object in the '703 patent) changes, change notification is selectively dispatched to the subscribed users interested in the event. The Cohen reference discloses how subscribing users express interest in specific events. When the target object changes, the Event Manager triggers the action (change notification) only for the specific subscribed users who expressed interest in that target object. Thus the Event Manager selectively dispatches the change events: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. *Pg. 210 (Emphasis added.* The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210(Emphasis added)..* **9.1. Object-Oriented Event Handling** … On a subscription, the Event Manager passes the condition on to the type manager without any interpretation; each type manager determines the sort of events it will monitor and interprets the condition accordingly. When an event is detected, the type manager signals the Event Manager, which then triggers the action associated with the subscription. The action is executed under the authority of the subscribing user. *Pg. 211 (Emphasis added).* The Cohen reference also describes a second disclosed means for selectively dispatching change events. The Event Manager triggers the action (sending notification to the subscribers) only if the corresponding subscribers still have the right to access the targeted object. This implementation of change notification thus allows the Event Manager to selectively dispatch change events: Before triggering the action, the Event Manager first checks the authority list of the targeted object to ensure that the subscriber still has the right to access it. If not, the subscription is canceled and no action is triggered. … Type managers which perform additional access control based on internal state (such as the Version Manager) will need to retain the identity of the subscriber internally, and when an event occurs, check internally whether the subscriber still has the right to monitor it. *Pg. 211 (Emphasis added).*

8[e] receiving the notification by the at least one of the plurality of objects and taking action based on the notification.	This element is disclosed in the Cohen reference. The Gypsy Environment allowed for subscribed users to receive notification about changes to a target object and take actions based on the notification. In Gypsy, subscribing users (corresponding to the plurality of objects in the '703 patent) express an interest in specific events by making a subscription. At least one of the subscribing users that expressed an interest in the event can receive a notification of the event: **9. Event Management** The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs. Its functionality is similar to DSEE monitors [Leblang84b], however, Gypsy allows arbitrary programs to be executed as actions, and supports the inclusion of new types of events by individual type managers. The user expresses interest in an event by making a subscription, consisting of a target object, a condition, and an action. For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs. *Pg. 210(Emphasis added).* As noted above, as used in the Cohen reference, the term "user" could be reasonably understood to mean either: (1) a particular individual (e.g., a subscribing individual); or (2) an object (e.g., a program) associated with that particular individual. One skilled in the relevant field would understand that, within the context of a computer system such as the one described in Cohen, dispatching a notification to a particular individual would typically involve electronically sending the notification to a computing device associated with the individual, and that the individual's computing device would include a program (e.g., an object) for receiving the notification and taking action in response to receiving the notification (e.g., presenting the notification to the individual in an appropriate manner). In addition to receiving the notification, the subscribing individual (or, as would be understood by one skilled in the field, a program associated with the subscribing individual) can take various actions based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computing device that would serve to alert the subscribing individual that they have received notification of changes to an object that the individual is monitoring. The Cohen reference also describes various other actions that a subscribing individual (or a program associated with the subscribing individual) can take while monitoring changes to particular objects (e.g., versions of particular documents): We expect that a user will most often monitor the unlocking of a branch (so that it can lock it), and the release of a new version (so that it can be used in building a configuration). *Pg 211 (Emphasis added).* In a further example discussed below, Cohen teaches that a subscribing individual (or, as would be understood by one skilled in the field, a program associated with the subscribing individual) may automatically produce a new version of a document in response to the release of a new version of one of the document's chapters. Reading the Cohen reference as a whole, one skilled in the field would understand that this type of action would be taken in response to the subscribing individual receiving notification that the object has changed (e.g., that a new version of the document has been released). When objects are hierarchically organized, a user may want to version control both the top level object as well as the sub-objects. For example, if a document is made up of a collections of chapters, it may be useful to version control each of the chapters separately, and also maintain versions of the document as a whole. One method of coping with hierarchical versions is to percolate changes up the hierarchy; releasing a new version at one level automatically causes a release of a new version at each higher level [Zdonik86]. For example, releasing a new version of a chapter would automatically cause a new version of the entire document to be released.

	Hierarchical version management in Gypsy can be provided, more cleanly, and with finer control, through the use of configurations. If chapters are individually version controlled, then a document can be represented as a (version-controlled) configuration with each chapter as a component. A user can explicitly produce a new version of the document by making a new version of the configuration and binding its components. *Pg 212 (Emphasis added).*
9. A method as recited in claim 8, including the step of notifying a plurality of objects.	This element is disclosed in the Cohen article. The Gypsy Environment describes an Event Manager that provides support for monitoring changes to objects and notifying a plurality of objects when an event occurs: The Gypsy Event Manager provides support for monitoring changes to objects, and triggering actions when an event occurs…. For the action, the subscriber may choose to either receive notification… *Pg. 210 (Emphasis added).* Thus, in the Gypsy Environment, the Event Manager can trigger a plurality of actions when an event occurs, each action corresponding to notification to a subscriber.
13. A method as recited in claim 8, including the step of removing an object as an action based on the notification.	This element is disclosed in the Cohen article. The Cohen article teaches triggering an action associated with a subscription upon the occurrence of a change event. "The Gypsy Event Manager provides support for monitoring changes to objects, and for triggering actions when an event occurs." *(Pg. 210).* The article broadly defines action: "For the action, the subscriber may choose either to receive notification, or to supply a program which will be executed when the event occurs." *Pg. 210 (Emphasis added).* The Cohen article further teaches "[a]n event may be a recurring event or a one-time event. A one-time event subscription is canceled when the event occurs." *Pg. 210 (Emphasis added).* In describing a one-time event subscription, the Cohen article effectively discloses the removal of an object as an action based on the notification. This is because in Gypsy, after a one-time event subscription is completed, the Event Manager triggers an action and the subscription is canceled, removing the object from future notifications.

The Cohen article discloses all of the elements of claims 1, 2, 6, 8, 9, and 13 of the '703 patent. Thus, claims 1, 2, 6, 8, 9, and 13 were anticipated by the Cohen article, which creates a substantial new question of patentability.

(b) Second Ground for Rejection: Claims 1-14 would have been obvious under 35 U.S.C. § 103 in light of the teachings of the Cohen article and the Bernstein patent.

Claim	Prior Art
1. An object-oriented notification framework system, comprising:	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here.
1[a] means for connecting a plurality of objects to a notification source;	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article to connect a plurality of subscriber objects to a target object.
1[b] memory means for storing connection information for the plurality of objects in a connection object of an object-oriented operating system;	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article to store connection information in a connection object via a memory means in a computer system.
1[c] means for registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
1[d] means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system; and	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
1[e] means for at least one of the plurality of objects to receive the notification and take action based on the notification.	The analysis for claim 1 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
2. A system as recited in claim 1, including processor means for notifying a plurality of objects.	The analysis for claim 2 provided in the First Ground for Rejection is incorporated here. The Cohen reference clearly discloses the notification of at least one subscriber. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would be obvious to one skilled in the art to further implement a system where a plurality of objects are notified of a change event.

3. A system as recited in claim 1, including processor means for changing a color of an object as an action based on the notification	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of changing the color of an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (4) highlight areas which are also transparent in that the patterns of the underlying data will be discernible, but the colors of the underlying data will be changed (also sometimes known as reverse video, e.g., all underlying black color is changed to white, all white color is changed to black, all blue color is changed to yellow, etc.), and (5) invisible which are truly invisible with regard to any occlusion of the underlying data. (Col. 2:33-51). In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to change the color of an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring
4. A system as recited in claim 1, including processor means for highlighting an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of highlighting an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (3) highlight frames which, like black frames, provide for transparency, but which have frames which are guaranteed to be visible (particularly useful compared to black frames when some of the underlying data may be black or very dark), (4) highlight areas which are also transparent in that the patterns of the underlying data will be discernible…" (Col. 2:33-45) In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to highlight an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.
5. A system as recited in claim 1, including processor means for reverse videoing an object as action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of reverse videoing an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (4) highlight areas which are also transparent in that the patterns of the underlying data will be discernible, but the colors of the underlying data will be changed (also sometimes known as reverse video, e.g., all underlying black color is changed to white, all white color is changed to black, all blue color is changed to yellow, etc.), and (5) invisible which are truly invisible with regard to any occlusion of the underlying data. (Col 2:33-51)

	In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to reverse video an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.
6. A system as recited in claim 1, including processor means for removing an object as an action based on the notification.	The analysis for claim 6 provided in the First Ground for Rejection is incorporated here. Also, as discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. Cohen specifically teaches allowing users to monitor when particular objects (versions of a particular document) are destroyed. "The Version Manager supports subscriptions to events for version group objects. It supports a variety of conditions, which effectively allow monitoring of events such as: • When a specified operation (e.g., release, destroy, change property) is taken on any version satisfying a specified predicate)." (Pg. 211) One skilled in the art would have understood that one way of indicating that an object (e.g., a version of a particular document) has been destroyed would be to remove a corresponding object from display on an individual's computer display screen. Accordingly, to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article and the Bernstein patent.
7. A system as recited in claim 1, including processor means for opening a window associated with an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein article further describes a "windowing computer display system in which data from one or more applications may be displayed in separate windows." (Col. 33:38 – Col. 34:1) In light of the above, one skilled in the art would have understood that one way of visually alerting an individual to the presence of a notification would be to open a new window on a display screen of a computer associated with the individual. Accordingly, to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article and the Bernstein patent.
8. A method for implementing an object-oriented notification framework system, comprising the steps of:	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here.

8[a] connecting a plurality of objects to a notification source;	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
8[b] storing connection information for the plurality of objects in a connection object of an object-oriented operating system;	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
8[c] registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here. Thus to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
8[d] selectively dispatching notification to at least one of the plurality of objects based on the connection registration information stored in the connection object of the object-oriented operating system; and	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
8[e] receiving the notification by the at least one of the plurality of objects and taking action based on the notification.	The analysis for claim 8 provided in the First Ground for Rejection is incorporated here. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
9. A method as recited in claim 8, including the step of notifying a plurality of objects.	The analysis for claim 9 provided in the First Ground for Rejection is incorporated here. The Cohen reference clearly discloses the notification of at least one subscriber. To the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would be obvious to one skilled in the art to further implement a system where a plurality of objects are notified of a change event.
10. A method as recited in claim 8, including the step of changing a color of an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of changing the color of an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (4) highlight areas which are also transparent in that the patterns of the underlying data will be discernible, but the colors of the underlying data will be changed (also sometimes known as reverse video, e.g., all underlying black color is changed to white, all white color is changed

	to black, all blue color is changed to yellow, etc.), and (5) invisible which are truly invisible with regard to any occlusion of the underlying data. (Col. 2:33-51). In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to change the color of an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.
11. A method as recited in claim 8, including the step of highlighting an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of highlighting an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (3) highlight frames which, like black frames, provide for transparency, but which have frames which are guaranteed to be visible (particularly useful compared to black frames when some of the underlying data may be black or very dark), (4***)*** highlight areas which are also transparent in that the patterns of the underlying data will be discernible…" (Col. 2:33-45) In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to highlight an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.
12. A method as recited in claim 8, including the step of reverse videoing an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein patent discloses the example of reverse videoing an object for a graphical user interface: "LMS provides link markers that can have many different appearance styles, including … (4) highlight areas which are also transparent in that the patterns of the underlying data will be discernible, but the colors of the underlying data will be changed (also sometimes known as reverse video, e.g., all underlying black color is changed to white, all white color is changed to black, all blue color is changed to yellow, etc.), and (5) invisible which are truly invisible with regard to any occlusion of the underlying data. (Col 2:33-51) In light of the teachings of the Cohen article and the Bernstein patent, it would have been obvious to one skilled in the art to reverse video an object as the action based on the notification. This would be done, for example, to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.
13. A method as recited in claim 8, including the step of removing an object as an action based on the notification.	The analysis for claim 13 provided in the First Ground for Rejection is incorporated here. Also, as discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring.

	Cohen specifically teaches allowing users to monitor when particular objects (versions of a particular document) are destroyed. "The Version Manager supports subscriptions to events for version group objects. It supports a variety of conditions, which effectively allow monitoring of events such as: • When a specified operation (e.g., release, destroy, change property) is taken on any version satisfying a specified predicate)." (Pg. 211) One skilled in the art would have understood that one way of indicating that an object (e.g., a version of a particular document) has been destroyed would be to remove a corresponding object from display on an individual's computer display screen. Accordingly, to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.
14. A method as recited in claim 8, including the step of opening a window associated with an object as an action based on the notification.	As discussed above in regard to claim element 1(e) in this same ground for rejection, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to notify the subscribing individual of the relevant changes to an object that the individual is monitoring. The Bernstein article further describes a "windowing computer display system in which data from one or more applications may be displayed in separate windows." (Col. 33:38 – Col. 34:1) In light of the above, one skilled in the art would have understood that one way of visually alerting an individual to the presence of a notification would be to open a new window on a display screen of a computer associated with the individual. Accordingly, to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article and the Bernstein patent.

The Cohen article discloses all of the elements of claims 1 and 8 of the '703 patent through an object-oriented change notification system. In Cohen, subscribing users express interests in target objects through the use of an Event Manager. Upon a change to a target object, the Event Manager triggers the action registered with the subscribing user. An action can either be a notification to the subscribing user, or execution of arbitrary programs when the event occurs. Cohen further teaches that a subscribing user (which one skilled in the art would understand to be either an individual or a computer program – e.g., an object) would take various actions in response to receiving a notification that a change has been made to a target object. For

example, the user may unlock a branch, or release a new version of an object (e.g., a particular document).

One skilled in the art would understand that, within the context of a computer system such as the one described in Cohen, dispatching a notification to a particular individual would typically involve electronically sending the notification to a computing device associated with the individual, and that the individual's computing device would include a program (e.g., an object) for receiving the notification and taking action in response to receiving the notification (e.g., communicating the notification to the individual in an appropriate manner).

As discussed in Cohen, the subscribing individual (or, as would be understood by one skilled in the art, a program associated with the subscribing individual) can take a variety of actions based on the notification. For example, one skilled in the art would understand that, in such situations, a program associated with the subscribing individual may initiate or modify a visual display of a particular object on the subscribing individual's computer, which would serve to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring. Accordingly, through this system, the Cohen article discloses all elements of claims 1 and 8 of the '703 patent.

Dependent claims 2 and 9 of the '703 patent require notifying a plurality of objects. Cohen teaches this through triggering one or more arbitrary programs upon occurrence of a change event. However, to the extent the patent owner may argue these elements are not explicitly disclosed by the Cohen article, they would have been obvious to one skilled in the art in light of the teachings of Cohen.

Dependent claims 6 and 13 require the action of removing an object as an action based on the notification. The Cohen article teaches this element through "one time events," where subscriptions are canceled when an event occurs.

Also, as discussed above, the Cohen article teaches sending a notification to a user (e.g., a subscribing individual or a program associated with the subscribing individual) and triggering an action based on the notification. For example, one skilled in the art would understand that, upon receiving the notification, a program associated with the subscribing individual may initiate or modify a visual display on the subscribing individual's computer that would serve to alert a subscribing individual that they have received notification of changes to an object that the individual is monitoring.

Cohen specifically teaches allowing users to monitor when particular objects (versions of a particular document) are destroyed. One skilled in the art would have understood that one way of indicating that an object (e.g., a version of a particular document) has been destroyed would be to remove a corresponding object from display on an individual's computer display screen. Accordingly, to the extent the patent owner may argue this claim element is not explicitly disclosed by the Cohen article, it would have been obvious to one skilled in the art in light of the teachings of the Cohen article.

Dependent claims 3, 4, 5, 7, 10, 11, 12, and 14 all deal with triggering different actions as a result of the event notification. The actions include changing the color of an object, highlighting an object, reverse videoing an object, and opening a new window associated with the object. The Cohen article explicitly teaches triggering an action as a result of a change notification. The Cohen article leaves the definition for "action" open and broad, and includes several different examples of actions that may be taken in response to the notification.

Cohen teaches that the subscribing individual (or, as would be understood by one skilled in the field, a program associated with the subscribing individual) can take various actions based on the notification. For example, one skilled in the art would understand that, in such situations, a program associated with the subscribing individual may initiate or modify a visual display of a particular object on the subscribing individual's computer that would serve to alert the individual that they have received notification of changes to an object that the individual is monitoring.

To the extent the patent owner may argue that the Cohen reference does not explicitly disclose the obvious and common sense actions of "changing color," "highlighting," "reverse videoing," and "opening a window for" an object, these actions would have been obvious in light of the combined teachings of Cohen and the Bernstein patent. The Bernstein patent explicitly discloses the actions of changing the color of, reverse videoing, and highlighting objects, and also opening new windows within the context of a graphical user interface. In light of the combined teachings of Cohen and Bernstein, it would have been obvious to modify objects displayed on an individual's display screen by "changing color," "highlighting," "reverse videoing," and "opening a window for" an object in order to alert the individual that they have received notification of changes to an object that the individual is monitoring. Thus, the Cohen article and the Bernstein patent create a substantial new question of patentability.

III. Conclusion

The Cohen article and Bernstein patent were not previously considered and they are not cumulative with the references previously considered. Consideration of obvious combinations of these references in accordance with the proposed Grounds for Rejection leads to the conclusion that these references create substantial new questions of patentability for claims 1-14 of the '703 patent. The Requester further submits that claims 1-14 must be rejected as unpatentable.

Respectfully submitted,

/Scott E. Brient/

Scott E. Brient

Reg. No. 44,561

Customer No. 93485
Brient Intellectual Property Law, LLC
2300 Lakeview Parkway, Suite 700
Alpharetta, GA 30009
Phone: (678) 916-3828
Fax: (678) 835-9310

ELECTRONICALLY FILED USING THE EFS-WEB ELECTRONIC FILING SYSTEM OF THE UNITED STATES PATENT & TRADEMARK OFFICE ON April 28, 2010.

EXHIBIT 5

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/010,967	04/28/2010	5315703	0919/01032	8070

7590 07/14/2010

JAMES A. WARD, PATENT COUNSEL OBJECT
TECHNOLOGY LICENSING CORPORATION
1 INFINITE LOOP
MAIL STOP 38-OTL
CUPERTINO, CA 95014

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 07/14/2010

Please find below and/or attached an Office communication concerning this application or proceeding.

Order Granting / Denying Request For Ex Parte Reexamination	Control No.	Patent Under Reexamination	
	90/010,967	5315703	
	Examiner	Art Unit	
	Christina Y. Leung	3992	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

The request for *ex parte* reexamination filed 28 April 2010 has been considered and a determination has been made. An identification of the claims, the references relied upon, and the rationale supporting the determination are attached.

Attachments: a)☐ PTO-892, b)☒ PTO/SB/08, c)☐ Other: _____

1. ☒ The request for *ex parte* reexamination is GRANTED.

RESPONSE TIMES ARE SET AS FOLLOWS:

For Patent Owner's Statement (Optional): TWO MONTHS from the mailing date of this communication (37 CFR 1.530 (b)). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).**

For Requester's Reply (optional): TWO MONTHS from the **date of service** of any timely filed Patent Owner's Statement (37 CFR 1.535). **NO EXTENSION OF THIS TIME PERIOD IS PERMITTED.** If Patent Owner does not file a timely statement under 37 CFR 1.530(b), then no reply by requester is permitted.

2. ☐ The request for *ex parte* reexamination is DENIED.

This decision is not appealable (35 U.S.C. 303(c)). Requester may seek review by petition to the Commissioner under 37 CFR 1.181 within ONE MONTH from the mailing date of this communication (37 CFR 1.515(c)). **EXTENSION OF TIME TO FILE SUCH A PETITION UNDER 37 CFR 1.181 ARE AVAILABLE ONLY BY PETITION TO SUSPEND OR WAIVE THE REGULATIONS UNDER 37 CFR 1.183.**

In due course, a refund under 37 CFR 1.26 (c) will be made to requester:

a) ☐ by Treasury check or,

b) ☐ by credit to Deposit Account No. _____, or

c) ☐ by credit to a credit card account, unless otherwise notified (35 U.S.C. 303(c)).

/Christina Y. Leung/ Primary Examiner, Art Unit 3992		

cc:Requester (if third party requester)

DECISION GRANTING EX PARTE REEXAMINATION

Decision on the Request

The present request for *ex parte* reexamination raises a substantial new question of patentability with respect to **claims 1-14** of United States Patent 5,315,703 to Matheny.

References Cited in the Request

Cohen (Cohen et al., "Version Management in Gypsy," ACM, pp. 201-215, 1988)

Bernstein (US 5,204,947 A)

Issues Raised by the Request

Issue 1

The request alleges that Cohen alone raises a substantial new question of patentability with respect to claims 1, 2, 6, 8, 9, and 13.

Issue 2

The request alleges that Cohen in view of Bernstein raises a substantial new question of patentability with respect to claims 1-14.

The Matheny Patent

The Matheny patent is generally directed to a system for an object based notification system. Claim 1 is representative:

1. An object-oriented notification framework system, comprising:

(a) means for connecting a plurality of objects to a notification source;

(b) memory means for storing connection information for the plurality of objects in a connection object of an object-oriented operating system;

(c) means for registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented operating system;

(d) means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system; and

(e) means for the at least one of the plurality of objects to receive the notification and take action based on the notification.

Prosecution History

Claims 1-14 are the current claims in the Matheny patent, which issued 24 May 1994 from application 07/996,782 filed 23 December 1992.

23 December 1992: Applicant originally filed claims 1-14.

27 July 1993: Examiner rejected claims 1, 2, 8, and 9 under 35 U.S.C. 102 as being anticipated by Microsoft Systems Journal ("Microsoft Systems Journal, A Presentation Manager Primer," vol. 5, no. 1, p. 14, January 1990); and claims 3-7 and 10-14 under 35 U.S.C. 103 as being unpatentable over Microsoft Systems Journal in view of Microsoft Windows User's Guide ("Microsoft Windows User's Guide," version 3, pp. 52 and 83-85, 1990-1992).

17 September 1993: Applicant amended claims 1-8 and added new claims 15-23.

13 December 1993: Examiner amended claims 3 and 10 to correct informalities and canceled claims 15-23 by Examiner's Amendment. Examiner allowed claims 1-14 without an explicit statement of reasons for allowance.

Detailed Analysis

Claims 1-14 will be reexamined. In view of the prosecution history, a substantial new question of patentability is raised by the evaluation of a prior art reference (or a combination of prior art references) that teaches the features and limitations added to the claims in the 17 September 1993 amendment. Specifically, these limitations include means for registering connection information, including registration information indicative of a notification status, in the connection object of the object-oriented system; and means for selectively dispatching notification to at least one of the plurality of objects based on the registration information stored in the connection object of the object-oriented system.

Issues 1 and 2

Cohen is new prior art. Cohen teaches, among other things, means for registering connection information including registration information indicative of a notification status (i.e., Cohen discloses that an action associated with an event may comprise a notification; page 210, "9. Event Management") Cohen also teaches means for selectively dispatching notification to at least one of the plurality of objects based on the registration information (i.e., Cohen discloses that an Event Manager triggers an action such as notification when an event is detected; pages 210-211, "9.1. Object-Oriented Event Handling"). Since these teachings are directly related to subject matter considered the basis for allowability of claims 1-14, a reasonable examiner would consider evaluation of Cohen important in determining the patentability of the claims. Therefore, Cohen, alone or in view of Bernstein, raises a substantial new question of patentability with respect to claims 1-14.

Conclusion

Extensions of time under 37 CFR 1.136(a) will not be permitted in these proceedings because the provisions of 37 CFR 1.136 apply only to "an applicant" and not to parties in a reexamination proceeding. Additionally, 35 U.S.C. 305 requires that *ex parte* reexamination proceedings "will be conducted with special dispatch" (37 CFR 1.550(a)). Extensions of time in *ex parte* reexamination proceedings are provided for in 37 CFR 1.550(c).

The patent owner is reminded of the continuing responsibility under 37 CFR 1.565(a) to apprise the Office of any litigation activity, or other prior or concurrent proceeding, involving Patent No. 5,315,703 throughout the course of this reexamination proceeding. The third party requester is also reminded of the ability to similarly apprise the Office of any such activity or proceeding throughout the course of this reexamination proceeding. See MPEP §§ 2207, 2282 and 2286.

All correspondence relating to this *ex parte* reexamination proceeding should be directed:

By Mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Any inquiry concerning this communication should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

/Christina Y. Leung/

Primary Examiner, Art Unit 3992

/D. M. H./
Primary Examiner, Art Unit 3992

ESK

Substitute for form 1449/PTO
(Revised 07/2007)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet	1	of	1

Complete if Known	
Application Number	07/996,782
Filing Date	Dec. 23, 1992
Patent Number	5,315,703 (Exhibit A)
Issue Date	May 24, 1994
First Named Inventor	Matheny et al.
Art Unit	2317
Examiner Name	D. Shaw
Attorney Docket Number	0919/01032

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document Number Number - Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages of Relevant Figures Appear
/CL/	1	US-5,204,947 (Exhibit C)	04-20-1993	Bernstein et al.	

OTHER DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s) , volume-issue number(s), publisher, city and/or country where published.	English Language Translation Attached
/CL/	2	COHEN et al., Version Management in Gypsy, 1988, Pages 201-215, ACM (Exhibit B)	

Examiner Signature	/Christina Leung/	Date Considered	07/08/2010

*Examiner: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

EXHIBIT 6

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Dilts
U.S. Patent No.:	5,455,854
Issued:	October 3, 1995
Group Art Unit:	2601
Serial No:	08/108,877
Examiner:	H. Hong
Filed:	October 26, 1993
For:	OBJECT-ORIENTED TELEPHONY SYSTEM
Attorney Docket No.	0919/01030

April 28, 2010

Mail Stop *Ex Parte* Reexamination
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

REQUEST FOR REEXAMINATION

Reexamination of United States Patent 5,455,854 (hereinafter, "the '854 patent"), which issued October 3, 1995 to Dilts is requested under 35 U.S.C. §§ 302-307, and under 37 C.F.R. § 1.510. This patent is still in force.[1] A copy of the patent in accordance with 37 C.F.R. § 1.510(b)(4) is submitted herewith as Exhibit A.

I. Claims for which Reexamination is Requested

The '854 patent discloses a method and apparatus according to which telephony elements are enabled by utilizing telephony objects which interface with and pass information to the telephony elements. The patent describes an object-oriented operating system that controls a telephony object containing data and logic. The logic of the telephony object is utilized to interface the telephony element to a processor by initiating a call connection, monitoring call progress, activating call features and storing status information

[1] Indeed, a counterclaim for alleged infringement of the '854 patent has been filed in the U.S. District Court for the District of Delaware, *Nokia Corp. v. Apple Inc.*, Case No. 1:09-cv-00791-GMS. That litigation is in its early stages and no discovery regarding the '854 patent has taken place. If the litigation proceeds, Third Party Requester expects there will be a challenge to the validity of the '854 patent therein.

in the data of the telephony object. Reexamination is requested of all Claims 1-24 of the '854 patent.

II. Statement of Substantial New Questions of Patentability

A. The Subject Matter of Claim 1-24

Claims 1-24 recite:

1. A telephony apparatus, comprising:

(a) a processor;

(b) a storage attached to and controlled by the processor;

(c) an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage and controlling operations of the processor;

(d) a display attached to the processor under the control of the object oriented operating system;

(e) a telephony element attached to the processor;

(f) a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and

(g) means for controlling the telephony element by the object oriented operating system utilizing the logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object.

2. The apparatus as recited in claim 1, including means for translating information received from the telephony element into information the object oriented operating system can utilize.

3. The apparatus as recited in claim 1, including means for translating information received from the telephony object into information the telephony element can utilize.

4. The apparatus as recited in claim 1, including means for attaching the telephony element to the processor.

5. The apparatus as recited in claim 4, including means for connecting a telephone line to the processor.

6. The apparatus as recited in claim 4, including means for connecting a handset to the processor.

7. The apparatus as recited in claim 4, including means for setting up a call to the processor.

8. The apparatus as recited in claim 4, including means for passing information between the telephony element and the processor.

9. The apparatus as recited in claim 8, including means for exchanging DTMF tones between the telephony element and the processor.

10. The apparatus as recited in claim 1, including means for enabling features of the telephony element via the telephony object.

11. The apparatus as recited in claim 1, including means for servicing queries between a telephony element and the object-oriented operating system.

12. The apparatus as recited in claim 1, including means for exchanging notification information between a telephony element and the object-oriented operating system.

13. A method for enabling telephony elements on a computer system, including a processor with an attached storage, display and telephony element, comprising the steps of:

(a) controlling operations of the processor with an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage;

(b) creating a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and

(c) controlling the telephony element by the object-oriented operating system utilizing logic in the telephony object to interface the telephony element to the processor by initiating

a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object.

14. The method as recited in claim 13, including the step of translating information received from the telephony element into information the object oriented operating system can utilize.

15. The method as recited in claim 13, including the step of translating information received from the telephony object into information the telephony element can utilize.

16. The method as recited in claim 13, including the step of attaching the telephony element to the processor.

17. The method as recited in claim 16, including the step of connecting a telephone line to the processor.

18. The method as recited in claim 16, including the step of connecting a handset to the processor.

19. The method as recited in claim 16, including the step of setting up a call to the processor.

20. The method as recited in claim 16, including the step of passing information between the telephony element and the processor.

21. The method as recited in claim 20, including the step of exchanging DTMF tones between the telephony element and the processor.

22. The method as recited in claim 13, including the step of enabling features of the telephony element via the telephony object.

23. The method as recited in claim 13, including the step of exchanging status information between a telephony element and the object-oriented operating system.

24. The method as recited in claim 13, including the step of exchanging notification information between a telephony element and the object-oriented operating system.

In reexamination, as with any proceeding before the U.S. Patent and Trademark Office ("USPTO"), the terms and phrases of a claim are given their broadest reasonable construction. *E.g., In re American Academy of Science Tech Center*, 367 F.3d 1359, 70 USPQ2d 1827, 1830 (Fed. Cir. 2004) ("During examination, 'claims ... are to be given their broadest reasonable interpretation . . .." (*quoting In re Bond*, 910 F.2d 831, 833, 15 USPQ2d 1566 (Fed. Cir. 1990))).

B. Newly cited Prior Art

The '854 patent matured from a U.S. patent application filed October 26, 1993, and claims no earlier priority filing date. Therefore, the "Critical Date" for prior art relevant to the claims of the '854 patent, under 35 U.S.C. § 102(b) is October 26, 1992.

The requestor respectfully submits that the prior art under §§ 102(b) and/or 102(a) taught or suggested the subject matter of the claims of the '854 patent. More particularly, the requester submits that:

- Anders Sixtensson, Wenchaun Ye, "Reuse In The Telecommunication Domain Using Object Oriented Technology And ADA" (Washington Ada Symposium Proceedings June 1990) pages 231-239 ("the Sixtensson article") (Exhibit B);

in view of:

- Ellis S. Cohen et al., "Version Management in Gypsy" (Siemens Research & Technology Laboratories 1988) pages 201-215 ("the Cohen article") (Exhibit C);

and further in view of:

- "NeXTSTEP™ OBJECT-ORIENTED PROGRAMMING AND THE OBJECTIVE C LANGUAGE", NeXTSTEP Developer's Library Release 3 (Addison-Wesley Publishing Company April 1993) pgs. xi – xvi, 1, 5, and 8-18 ("NeXTSTEP I") (Exhibit D);
- "NeXTSTEP™ PROGRAMMING INTERFACE SUMMARY", NeXTSTEP Developer's Library Release 3 (Addison-Wesley Publishing Company April 1993) pgs. 1, and 13-1 – 13-7 ("NeXTSTEP II") (Exhibit E);

- "NeXTSTEP™ GENERAL REFERENCE", Volume 2, NeXTSTEP Developer's Library Release 3 (Addison-Wesley Publishing Company November 1992) pgs. 13-1 – 13-38 ("NeXTSTEP III") (Exhibit F);

and further in view of:

- U.S. Patent No. 4,625,081 to Lotito et al. ("the Lotito patent") (Exhibit G);
- "Programming the Display PostScript® System with NeXTstep™" (Addison-Wesley Publishing Company November 1991) pgs. xxii, xxiii, 1, 2, 3, 4, 7 and 10 ("NeXTSTEP IV") (Exhibit H);

rendered the subject matter of the claims of the '854 patent obvious to one of ordinary skill in the relevant art. NeXTSTEP I, II, III, and IV are sometimes collectively referred to herein as "NeXTSTEP").

Furthermore, the requester notes that the Sixtensson article, the Cohen article, NeXTSTEP I, NeXTSTEP II, NeXTSTEP III, the Lotito patent and NeXTSTEP IV were not listed on the face of the '854 patent and are not cumulative to prior art considered. Consequently, the Sixtensson article, the Cohen article, NeXTSTEP I, NeXTSTEP II, NeXTSTEP III, the Lotito patent and NeXTSTEP IV are newly applied and unquestionably raise substantial new questions of patentability.

C. Basis for Substantial New Questions of Patentability

Claims 1-24 of the '854 patent do not patentably distinguish over combinations of the above-noted newly cited references. In summary, the Sixtensson article discloses a software system for providing a telephony service interface between a processor and telephony elements such as handsets, telephone lines, and the like. For this purpose, the Sixtensson article discloses an overall object-oriented method for designing objects in an object-oriented

language (OSDL) that is translated for usage into the Ada language. The method is formulated based on using Plain Ordinary Telephony Service (POTS) with a number of new service features for telecommunication applications on a prototyping system that is suitable for general use. Prior to translating its object-oriented program design into Ada for usage, the Sixtensson article discloses usage of System Interactive Diagrams (SIDs) and Object Interactive Diagrams (OIDs) representative of telephony objects that have data and logic for interfacing telephony elements. (Exhibit B, pg. 231, 232, 233).

The Cohen article discloses a Gypsy programming support environment in which Gypsy is built on top of an object-oriented operating system that includes objects. (Exhibit C, pgs. 201 and 202). The Gypsy programming environment would have been suitable for implementing the Sixtensson object-oriented approach to providing a telephony service interface in an object-oriented programming language.

NeXTSTEP I discloses mechanisms related to object-oriented programming. In particular, NeXTSTEP I discloses usage of applications and operating systems in the same object-oriented programming environment and programming language, Objective C. (Exhibit D, p. xii). NeXTSTEP II discloses use of applications and operating systems in the same object-oriented program language for software kits such as, for example, phone kits. (Exhibit E, pgs. 13-1 – 13-7). NeXTSTEP III teaches utilization of an objected oriented programming language such as Objective C for applications related to telephony, for example the Plain Old Telephone Service (POTS) and Integrated Services Digital Networks (ISDNs). (Exhibit F, pg. 13-3). NeXTSTEP III also discloses use of software kits such as, for example, phone kits in an object-oriented program language, Objective C. (Exhibit F, pgs. 13-6 – 13-14).

The Lotito patent discloses an automated telephony system having an operating system in which the applications and the operating system are programmed in the same programming language. For example, the Lotito patent discloses that the applications and the operating system may both be implemented in Pascal. (Exhibit G, col. 75, lns. 39-43, col. 94, lns. 47-51, and col. 23, lns. 13-14, 60-61).

Thus, the Sixtensson article discloses or would have rendered obvious each element of claims 1- 24 of the '854 patent, except for the final computer code including an object-oriented operating system and interface, and certain additional dependent claim limitations discussed below. The Cohen article discloses an object-oriented operating system in which objects are utilized in the object-oriented operating system. One skilled in the art would have found a clear motivation to modify the Sixtensson article with the teachings of the Cohen article given that Cohen touts the benefits of a programming support environment that facilities usage of a programming language in an object-oriented operating system that includes objects and Sixtensson touts the benefit of object-oriented program design. The result of applying such teaching from the Cohen article to the Sixtensson article method was predictable, namely, on using an overall object-oriented method to extend automated telephony service features in an object-oriented operating system, Sixtensson's automated telephony interface would have provided the same functions while benefitting from the advantages of an object-oriented operating system.

Additionally, NeXTSTEP I discloses an object-oriented operating system and object-oriented applications written in the same programming language. NeXTSTEP II and NeXTSTEP III disclose object-oriented applications for use in telephony. One skilled in the art would have found a clear motivation to modify the Sixtensson article with the teachings

of NeXTSTEP I, NeXTSTEP II and NeXTSTEP III, given that NeXTSTEP I, NeXTSTEP II and NeXTSTEP III tout the benefits of using object-oriented applications and object-oriented operating systems in telephony, and Sixtenssion similarly touts the benefit of object-oriented program design. The result of applying such teaching from NeXTSTEP I, NeXTSTEP II and NeXTSTEP III to the Sixtensson article method was predictable, namely, on using an object-oriented programming language for applications as opposed to translating objects into Ada, Sixtensson's automated telephony interface would have provided the same functions while benefitting from the advantages of an object-oriented operating system and object-oriented applications in the same programming language. Indeed, the combination of NeXTSTEP I, II and III teachings with the Sixtensson article may be viewed either as applying a known technique (an object-oriented programming environment) to a known method or product (automated telephony) ready for improvement to yield predictable results[2], or as a case of known work in one field of endeavor (NeXTSTEP applied to automate telephony) prompting variations (Sixtensson object design for automated telephony) of it for use in the same field based on design incentives (extending object-oriented programming benefits) where the variations are predictable to one of ordinary skill in the art.[3]

The Lotito patent's disclosure of using applications and an operating system in the same programming language for a telephony system provides further motivation to modify the operating system of Sixtensson to be an object-oriented operating system.

Moreover, the disclosure of NeXTSTEP IV, which relates to a programming environment in which an object-oriented programming language such as Objective-C is used as the main programming language in a Display PostScript system provides additional

[2] MPEP § 2143 (D) Examples of Basic Requirements of a *Prima Facie* Case of Obviousness.
[3] MPEP § 2143 (F).

motivation to modify the operating system of Sixtensson to be an object-oriented operating system.

The examiner in the original prosecution of U.S. Patent Application Serial No. 08/108,877 cited as reasons for allowance over four pages of arguments by the applicant. These arguments focused on the absence in the cited prior art combination of a computer system using "objects" to perform telephony processes and control a telephony element by an object-oriented operating system. (Response Under 37 C.F.R. § 1.116 filed June 20, 1995, pages 2-7). The gap in the prior art subject matter described by these arguments is clearly met by the obvious implementation of Sixtensson's object-oriented design using the Cohen or the NeXTSTEP object-oriented operating system and programming environment including the Objective-C programming language to provide telephony objects.

Just as the Office has taken pains to assure that patents are not granted (or, presumably, reaffirmed on reexamination) for inventions that merely implement known concepts on the Internet[4], the Office should not affirm a patent that merely implements known concepts using a known alternative programming environment. The claims of the '854 patent present such a case, because they claim implementing the known automation of telecommunications features by merely substituting known object-oriented programming to provide both an operating system and applications support for a system providing those known telecommunications features.

[4] See, FORMULATING AND COMMUNICATING REJECTIONS UNDER 35 U.S.C. 103 FOR APPLICATIONS DIRECTED TO COMPUTER-IMPLEMENTED BUSINESS METHOD INVENTIONS, IV.B. Examples 9,10 (USPTO Paper at http://www.uspto.gov/patents/resources/methods/busmeth103rej.jsp#E9)

Because the Sixtensson article, the Cohen article, NeXTSTEP I, NeXTSTEP II,[5] NeXTSTEP III, NeXTSTEP IV and the Lotito patent were not previously considered and are not cumulative of any reference previously considered, combination of these references necessarily raises new, substantial, and not cumulative, questions of patentability. Consequently, Reexamination of claims 1-24 of the '854 patent must be ordered and the claims rejected.

D. Application of Prior Art References to Claims 1-24

1. Content of the Prior Art

(a) The Sixtensson Article

The Sixtensson article discloses an object-oriented method for extending Plain Ordinary Telephony Service (POTS) with a number of new service features on a prototyping system. In this regard, the Sixtensson article touts that "Object oriented Software Engineering (OOSE) is the most promising technique … for extensibility and reuse" and describes that object-oriented Domain Analysis (OODA) has been found vital for the success of reuse." (pg. 231) In order to extend POTS with a number of new service features, the Sixtensson article discloses generating an object-oriented model using System Interactive Diagrams (SIDs) and Object Interactive Diagrams (OIDs) which are associated with objects. The objects are used by an Object-oriented SDL (OSDL) implementation language so that the system of Sixtensson may translate the objects to an Ada implementation language.

The Sixtensson article discloses a telephony apparatus having a processor, and a storage attached to and controlled by the processor. In this regard, the Sixtensson article

[5] Third Party Requester points out that NeXTSTEP I and NeXTSTEP II have the same publication date, were meant to be used together, and may properly be considered as a single reference.

discloses that the overall object-oriented method used when extending the POTS software is performed on a prototyping system that includes a Line Interface Module (LIM) from an MD 110 PABX system by Ericsson and a SUN 3/50 workstation. (pg. 232) The Sixtensson article explains that the LIM of the prototyping system is a *micro processor* controlled unit that can connect a number of subscriber lines, extension lines, truck lines and operator lines. (p. 232) The SUN 3/50 workstation has "call handling software" that is implemented and executed. (pg. 232) As such, a person having ordinary skill in the art would understand that a SUN 3/50 workstation has a storage such as a memory and that the LIM may control the storage of the SUN 3/50 workstation. Page 232 of the Sixtensson article describes that the "[t]hough the … work relied on the prototyping system, the method … is of a general nature." As such, a skilled artisan would understand that the software design of Sixtensson could be applied and used in a real world system. The memory is shown diagrammatically in FIG. 1 of Sixtenssion, reproduced below.

The Sixtensson article also discloses an operating system, supporting encapsulation, polymorphism, and inheritance, which includes objects. Page 232 of the Sixtensson article describes that the SUN 3/50 workstation implements software such as "call handling software." As such, skilled artisans would have understood that it is inherent that the SUN 3/50 workstation has an operating system. The Sixtensson article teaches an operating system supporting encapsulation and inheritance based on at least the following:

> "The **encapsulation** of data and its operations facilitate the updating of the internal implementation of the operations working on the data. Without encapsulation, a search through the whole system is necessary to find all (perhaps) functions that operates on the actual data." (pg. 236) (emphasis added)
>
> "The **inheritance** on the subsystem level is achieved with the help of multi sublibraries." (pg. 234)

> A specific object telephone is described in the model by **inheriting** the common properties from the object terminal specialized with specific characteristics. (pg. 235)
>
> The other objects which were not updated, i.e. not recompiled, were **inherited** from pots.sub during linking." (pg. 237) (emphasis added)

On page 236, the Sixtensson article discloses the following:

> "Another powerful reuse mechanism is based on interchangeable parts. These interchangeable parts are objects when using OOSE. By carefully examining the functionality of each part, and by having well defined interfaces among the parts, the reuse of each part is greatly enhanced. For example, POTS and all the extension software systems use object telephone. There are many versions of source code describing object telephone. They are all interchangeable." (pg. 236)

A person with ordinary skill in the art would have understood that reuse mechanisms utilizing interchangeable parts discloses polymorphism. As such, Sixtensson teaches an operating system supporting polymorphism.

As shown in FIG. 3 of Sixtensson, set forth below for convenience, the Sixtensson article teaches that the prototyping system uses OIDs having "objects … such as "telephoneA, toneDevice, switch, call etc." and describes "**operations of the objects**." (pg. 233) (emphasis added) Although the Sixtensson article does not explicitly specify an operating system it is inherent that the SUN 3/50 workstation has an operating system in which objects are being used.

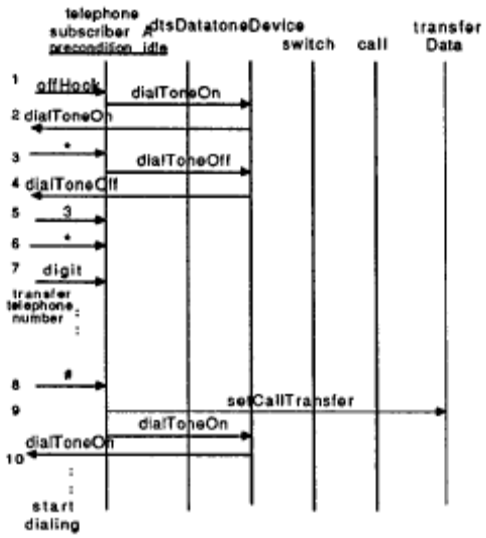


Figure 3: *Example of an OID*

Sixtensson, Figure 3

Moreover, Applicant's Admitted Prior (APA) found at col. 5. lines 34-37 of the '854 patent states: "As will be understood by those skilled in the art, Object-Oriented Programming (OOP) objects are software entities comprising **data structures** and **operations on the data**." (emphasis added) As such, skilled artisans would understand that the Sixtensson article teaches that each of the objects therein contain logic and data. Since the objects contain data and logic it is inherent that the objects reside in storage and that the logic controls operations of a processor that execute the software entities.

As shown in FIG. 1, set forth below for convenience, of the Sixtensson article, the SUN 3/50 workstation has a display attached to the LIM which is a "micro processor controlled unit" under the control of an operating system. (pg. 232) As also shown in FIG. 1 of the Sixtensson article, telephone T1 is attached to the LIM which is a "micro processor." (pg. 232) Thus, Sixtensson teaches a telephony element attached to the processor.

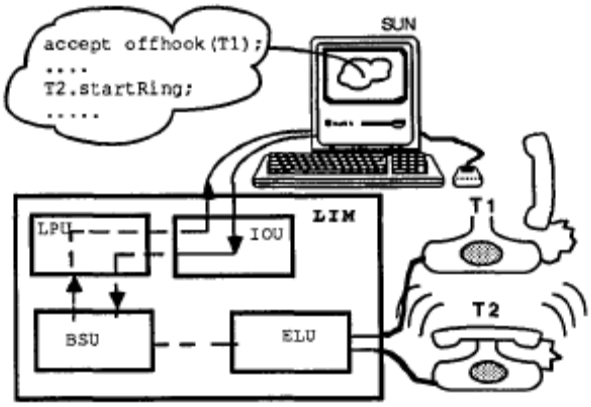


Figure 1: *The prototyping system*

Sixtensson, Figure 1

Sixtensson teaches that the objects which include logic may be telephony objects, which is supported by the following passages:

> "The ... objects decomposed from the system, such as **telephoneA**, **toneDevice**, switch, call etc." (pg. 233) (emphasis added)

> "Within the domain of telecommunication … typical objects are … **telephones**, **lines, tonedevices**." (pg. 234)

> "[A]n object called **terminal** which is an abstraction of different **subscriber equipments** such as **telephone, telex, telefax** etc." (pg. 235) (emphasis added)

As shown in FIG. 1, the telephony object "offhook" includes operations or logic for interfacing the telephony element telephone T1 to the LIM which is a "micro processor controlled unit" and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation. Furthermore, the Sixtensson article teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object. For instance, the telephony object (e.g., "tonedevices", dtsDatatoneDevice) stores status information associated with telephony element telephone T1, for example "offhook (T1)" when telephone T1 is off hook, as shown in FIG. 1.

The Sixtensson article teaches that the SUN 3/50 workstation controls the telephony element telephone 1 and "the objects telephoneA, toneDevice, switch, call" have operations

to interface the telephony element such as telephone T1 to the LIM, which is a micro processor, by initiating offhook, dialToneOn, dialToneOff, and setCallTransfer features.

(b) The Cohen Article

The Cohen article teaches an object-oriented operating system including objects. For instance, the Cohen article discloses a Gypsy programming support environment in which Gypsy is built on top of an object-oriented operating system that includes objects. (pgs. 201 & 202) The Cohen article discloses that Gypsy is "used to store, organize and selectively retrieve versions of objects" and "provides support for monitoring changes to objects." (pg. 201) Additionally, the Cohen article discloses that descriptors for all of a version group's … objects … are owned by Gypsy." (pg. 213) As such, the Cohen article discloses that the Gypsy programming support environment utilizes objects.

The Cohen article also discloses that Gypsy is built on an extension of Unix "that provides mechanisms for … **extensions** … of files and directories." (pg. 202) (emphasis added) By using these mechanisms, the Cohen article teaches that Gypsy version management features may be provided outside of the operating system kernel. (pg. 202) For instance, the Cohen article describes that Gypsy provides the ability to extend a file system to support reading a version controlled file without reconstituting the file by using "object-oriented extension mechanisms to keep the code out of the kernel." (pg. (209) In this regard, the Cohen article describes that Gypsy is integrated with the object-oriented operating system. (pgs. 201 & 202)

The Cohen article discloses that the objects of the object-oriented operating system contain logic and data. For instance, the Cohen article teaches that the objects include "**code**" that may be "invoked by calling a '**method**' (a.k.a. **procedure**)." (pg. 202)

(emphasis added) The Cohen article also discloses that the objects are accessible "by descriptors which carry control rights" and explains that a descriptor may be retrieved "**from an object**." (pg. 202) (emphasis added) Moreover, the Cohen article describes that "the … **contents of** … **objects**" may be used in queries. (pg. 203) Additionally, the Cohen article discloses that the "**binding of methods to code** is implemented through attributes" and describes that "an attribute is a list of methods used to implement the behavior." (pg. 202) (emphasis added)

The Cohen article teaches that the object-oriented operating system supports inheritance. For instance, the Cohen article discloses that the "LINK attribute, provid[es] the same masquerading as [a] multiple **inheritance** model." (pg. 213) (emphasis added)

The Cohen article also teaches that the objects containing logic and data are resident in storage. For instance, page 203 of the Cohen article discloses that properties of the objects "are represented in **data** structures that are laid out in the **memory**." (emphasis added)

(c) NeXTSTEP I, NEXTSTEP II, NeXTSTEP III & NeXTSTEP IV

The NeXTSTEP I reference teaches an object-oriented development environment that consists of libraries of objects, software kits, a set of development tools and an object-oriented programming language such as Objective C. NeXTSTEP I describes that Objective C is an object-oriented programming language. (pg. xi) In particular, NeXTSTEP I teaches an object-oriented operating system and object-oriented applications written in the same programming language and the following passage illustrates this functionality:

> "This book is about the third component of the development environment–the programming language. All NeXTSTEP software kits are written in the Objective C language. To get the benefit of the kits, **applications must also use Objective C**." (emphasis added)

Admitted Prior Art (APA) of the '854 patent teaches that a NeXTStep App Kit is part of an operating system (col. 7) and the quote above on pg. xii of NeXTSTEP I discloses that users in the prior art developed systems using the same NeXTStep system for an object-oriented operating system as well as object-oriented applications.

Moreover, the original assignee, Taligent, Inc., that prosecuted U.S. Patent Application Serial No. 07/996,782 (which issued as U.S. Patent No. 5,315,703) is the same assignee that prosecuted U.S. Patent Application Serial No. 08/108,877 (which issued as the '854 patent). On page 8 of an Amendment filed September 17, 1993 in U.S. Patent Application Serial No. 07/996,782, Taligent, Inc. stated that "**operating systems** such as … **NeXT** are UNIX based operating environments …" (emphasis added) As such, the assignee that prosecuted U.S. Patent Application Serial No. 08/108,877 admits as prior art that NeXTSTEP is an operating system.

NeXTSTEP I also discloses that object-oriented programming "groups operations and data into modular units called *objects*" and explains that "[e]very object has both state (data) and behavior (operations on data)." (pg. 1) NeXTSTEP I also teaches that "[a]n object is a group of related functions and a data structure that serves those functions" and describes that the functions are known as the object's *methods*." (pg. 5) Moreover, NeXTSTEP I discloses that "[o]bjects of the same kind … belong to the same *class*. (pg. 8) NeXTSTEP I also teaches an object-oriented programming language and object-oriented operating system supporting encapsulation, polymorphism and inheritance based on the following passages:

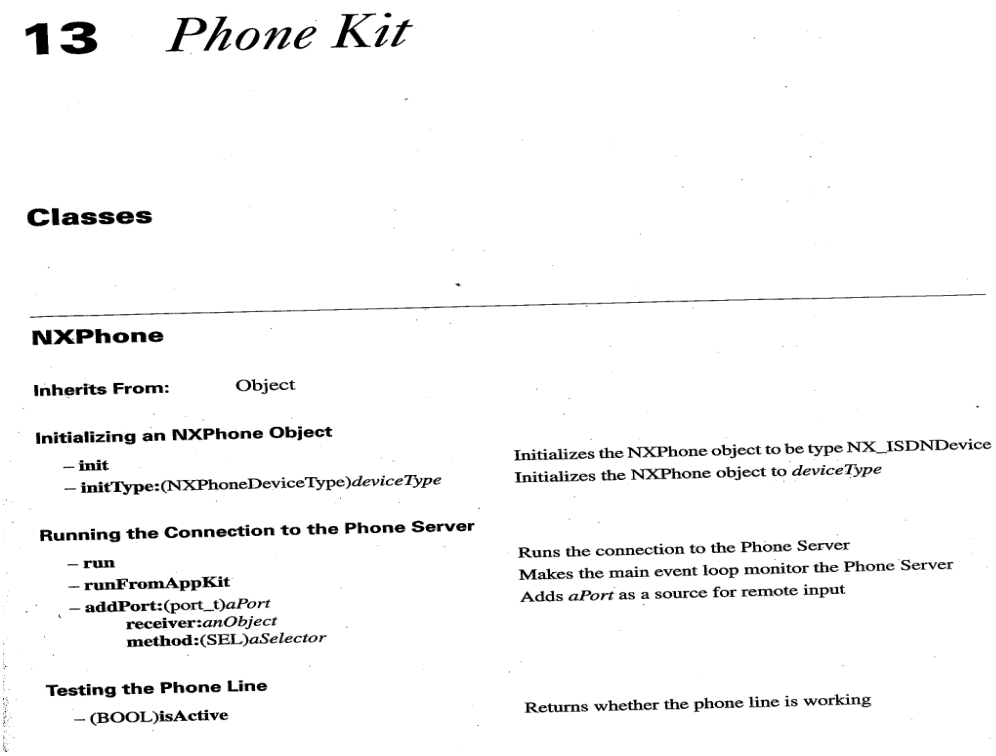
> "**Encapsulation** keeps the implementation of an **object** out of its interface, and **polymorphism** results from giving each **class** its own name space." (pg. 12) (emphasis added)

> Method implementations are similarly **encapsulated**, but, more importantly, so are an **object's** instance variables." (pg. 12) (emphasis added)

> "This ability of different **objects** to respond, each in its own way, to identical messages is called **polymorphism**." (pg. 13) (emphasis added)

> "[T]wo **classes** are connected so that the subclass ***inherits*** all the methods and instance variables of its superclass." (pg. 15) (emphasis added)

As shown below, NeXTSTEP II teaches that it is beneficial to use NeXTSTEP software kits in telephony.



13 *Phone Kit*

Classes

NXPhone

Inherits From: Object

Initializing an NXPhone Object

– **init** — Initializes the NXPhone object to be type NX_ISDNDevice
– **initType:**(NXPhoneDeviceType)*deviceType* — Initializes the NXPhone object to *deviceType*

Running the Connection to the Phone Server

– **run** — Runs the connection to the Phone Server
– **runFromAppKit** — Makes the main event loop monitor the Phone Server
– **addPort:**(port_t)*aPort* **receiver:***anObject* **method:**(SEL)*aSelector* — Adds *aPort* as a source for remote input

Testing the Phone Line

– (BOOL)**isActive** — Returns whether the phone line is working

NeXTSTEP II, pg. 13-1

The above annotated page of NeXTSTEP II also teaches a classified object that has member functions to implement telephony features.

Moreover, NeXTSTEP III discloses the following benefits of using NeXTSTEP software kits in telephony:

> "The **Phone Kit** offers an easy way for you to connect the application you're developing to a telephone line, to initiate and receive calls over the line, and to transmit and receive data during a call… The phone line can be **POTS (plain**

> **old telephone service)** line or an **ISDN (Integrated Services Digital Network)** line." (pg. 13-3) (emphasis added)

As shown below, NeXTSTEP III discloses classes in Objective C that have subclasses.

The Phone Kit consists of just three Objective C classes, all direct subclasses of the Object class:

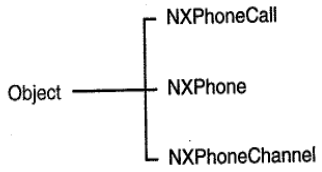


Figure 13-4. Phone Kit Inheritance Hierarchy

NeXTSTEP III, pg. 13-7

Pages 13-1 – 13-4 of NeXTSTEP II discloses that the NXPhoneCall, NXPhone and NXPhoneChannel subclasses each include various member functions.

Additionally, NeXTSTEP IV discloses a NeXT programming environment built around an object-oriented approach to software development in which an object-oriented programming language such as Objective-C is used as the main programming language. (pg. 1) NeXTSTEP IV also discloses Application Kits that are sets of predefined classes that can be incorporated within an application. (pgs. 1 & 3)

NeXTSTEP IV teaches objects containing logic and data at least based on Figure 1.1 and the following passages:

> "An object is a structure **containing data** and **functions** that operate on the data. (pg. 1) (emphasis added)
>
> **Data within an object** can be accessed … through the use of an **operator within the object**." (pg. 2) (emphasis added)
>
> "A unit of **data within an object** is called an instance variable. An **operation** is called a **method**." (pg. 2) (emphasis added)

NeXTSTEP IV also discloses that "[o]bjects with the same instance variable descriptions and methods are members of the same class" and describes that "[e]ach class has a class object." (pg. 2)

In addition, NeXTSTEP IV teaches that a message can be sent "from one object to another, instructing the receiver to perform an operation" and that "[d]ata necessary for the operation can be passed along with the message." (pg. 2) In this regard, NeXTSTEP IV teaches that "[a] message is a call to an object telling it to perform a particular method" and explains that "[a]rguments the method needs can be passed in the message." (pg. 7)

Page xxii of NeXTSTEP IV teaches that the system described therein supports encapsulation. Additionally, as shown in the following passages, NeXTSTEP IV discloses an object-oriented programming environment supporting inheritance.

> "A key feature of an object-oriented approach is **inheritance**. Objects are grouped in a hierarchy with descendants retaining data types and methods of the ancestor. As a result, a method defined several steps up in the hierarchy chain can be used to operate in a … distinct object." (pg. 2)

> "The top class in the Application Kit hierarchy is Object. It is the root for all Objective-C **inheritance** hierarchies, so all other classes **inherit** from it." (pg. 10)

> "Many of the major classes in the Application Kit **inherit** from the Responder class." (pg. 10)

In view of the foregoing, NeXTSTEP IV teaches the usage of applications and an operating system in the same programming language. As such, it would have been obvious to modify the Sixtensson article to utilize object-oriented applications and an object-oriented operating system in the same programming language in view of NeXTSTEP IV.

Given that NeXTSTEP IV discloses the benefits of using object-oriented applications in a programming environment and NeXTSTEP I, NeXTSTEP II and NeXTSTEP III disclose the benefits of using the object-oriented applications and an object-oriented operating system in the same programming language for telephony, it would have been obvious to modify the Sixtensson article to utilize object-oriented applications and an object-oriented operating system in the same programming language. Additionally, a skilled artisan would have been motivated to do so in view of Sixtensson's touting of the benefits of object-oriented techniques as it relates to telephony. (pgs. 231 & 232)

(d) The Lotito Patent

The Lotito patent 4,625,081, issued November 25, 1986, teaches a telephony system using an operating system in which the applications and the operating system are programmed in the same programming language:

> "**Application processing** or development is supported by the general purpose processors, which can directly execute the pseudo-machine codes, or P-codes, generated by a high level systems **programming language such as Pascal**" (col. 75, lns. 39-43) (emphasis added)
>
> "Level 3, user applications, represent **application programs** developed by users." (Col. 23, lns. 9-12) (emphasis added)
>
> "Major kernel system processes **implemented in Pascal**… This is necessary for those Level 2 system processes …" (Col. 94, lns. 47-51) (emphasis added)
>
> "Levels 0, 1 and 2 … are referred to as the kernel system," "The major processes of Level 2 system functions form the **operating system** of the **system 100** to which applications and other system processes make requests for services". (Col. 23, lns. 13-14, 60-63) (emphasis added)
>
> "**Extended Pascal** is the **language** supported on the **system 100**." (Col. 29, lines 25-26) (emphasis added)
>
> "**Processes** … are written in **extended Pascal**." (Col. 30, lines 13-14) (emphasis added)

> "A complete … application system utilizing all hardware/software resources of the **system 100** … realized as a set of programs coded in **Pascal**." (Col. 76, lines 4-6) (emphasis added)
>
> "A **level 3 Pascal**-programmed application task." (Col. 136, line 47) (emphasis added)

In view of the foregoing, the Lotito patent teaches the usage of applications and an operating system in the same programming language for telephony. As such, it would have been obvious to modify the Sixtensson article to utilize object-oriented applications and an object-oriented operating system in the same programming language in view of the Lotito patent.

Lotito also discloses telephony features for a telephone system in which a caller may access a message basket and leave a message that may be recorded. Upon entry of a code a caller may retrieve messages, playback the voice messages or forward the messages to other telephone lines. (Abstract & Col. 120, lines 143) Additionally, Lotito discloses that the telephone answering service of the telephone system provides the following features:

> "(1) Call intercept (Incall) handling, including message acquisition, voice, or typed.
> (2) Retrieval call (Recall) handling, including MFT and voice command control of playback.
> (3) Message dispatch (Outcall), the scheduled or unscheduled (demand) active delivery of a message" including forwarding messages to telephone numbers (mentioned often as a delivery option). (Col. 135, lines 51-58, Col. 3, lines 3-9 & Abstract)

As shown in the following passage Lotito also discloses activation of call features:

> "(1) Initialized … functions active ...
> (2) Ready to be activated …
> (3) Recording of playing back.
> (4) Pausing.
> (5) Terminating record or playback." (Col. 121, lines 3-8)

Lotito also discloses that the telephone system generates DTMF signals which are communicated through a telephone line. (Col. 12, lines 50-51) Additionally, Lotito discloses that the telephone system may "accept incoming calls as well as seize a line to dial outgoing calls and handle all DTMF … signals." (Col. 13, lines 18-20) Lotito also teaches that a real-time subsystem 230 can produce signals which can be applied to a channel and explains that in voice applications this is used to generate tones or tone sequences such as DTMF digit sequences. (Col. 119, lines 37-41)

2. Grounds for Rejection of the Claims

(a) ***First Ground for Rejection***: Claims 1-8, 10-20 and 22-24 would have been obvious over the Sixtensson article in view of the Cohen article.

By comparing the content of the Sixtensson article and the Cohen article to claims 1-8, 10-20 and 22-24 of the '854 patent, it will become clear to one of ordinary skill in the art that the claimed subject matter would have been obvious.

U.S. Patent No. 5,455,854	**Prior Art**
Claim 1	
A telephony apparatus, comprising:	**Sixtensson**, discloses a telephony apparatus (section 1, pg. 232: "the prototyping system [2] has been used repeatedly when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." See also FIG. 1
(a) a processor;	**Sixtensson** teaches a processor (section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) … LIM is a micro processor controlled unit.")
(b) a storage attached to and controlled by the processor;	**Sixtensson** discloses a storage attached to and controlled by the processor (FIG. 1 and section 2, pg. 232: A person having ordinary skill in the art would understand that a SUN 3/50 workstation includes a storage such as for example a memory and that the LIM which "is a micro processor controlled unit" may control the storage of the SUN 3/50 workstation by accessing the "serial ports".)

U.S. Patent No. 5,455,854	Prior Art
(c) an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage and controlling operations of the processor;	**Cohen** teaches an object-oriented operating system including objects (pgs. 201 & 202). **Cohen** teaches that each of the objects of the operating system contains logic and data (pg. 202: "[t]he code invoked by calling a 'method' (a.k.a. procedure)" & "[t]he binding of methods to code is implemented through attributes"). **Cohen** teaches an object oriented programming environment that extends software features for files and directories (pgs. 202 & 209). **Admitted Prior Art (APA) of the '854 patent** discloses known benefits of object oriented applications for communications (col. 6, lns. 14-20: frameworks are sets of object classes that collaborate to execute computing responsibilities; col. 6, lines 23-27 & col. 7, lns. 3-6: "developers … extend the framework" thereby extending the software application "and create customized solutions in a particular area of expertise" such as "frameworks that provide basic system software services such as **communications**." (emphasis added). These frameworks provide operating system level functions. **Sixtensson** teaches that it is beneficial to use an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232) & the **APA** discloses the benefits of object oriented frameworks that extend software services for communications. As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings of **Cohen**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 and section 4, pg. 235) **Cohen** teaches supporting inheritance (pgs. 202 & 213: "the LINK attribute, provid[es] the same masquerading as the multiple inheritance model.") **APA of the '854 patent** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lines 34-37) **Cohen** discloses each of the objects containing logic and data resident in the storage and controlling operations of the processor (pg. 203: "[t]o implement version selection … properties are represented in data structures that are laid out in the memory allocated to the … object").

U.S. Patent No. 5,455,854	Prior Art
(d) a display attached to the processor under the control of the object oriented operating system;	**Sixtensson** teaches a display attached to the processor under control of an operating system (FIG. 1: shows that the SUN 3/50 workstation includes a display attached to the LIM via "serial ports." The LIM is a micro processor. See also section 2, pg. 232). It would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **Cohen** as described above.
(e) a telephony element attached to the processor;	**Sixtensson** teaches a telephony element attached to the processor (FIG. 1: shows that telephones T1 and T2 are attached to LIM via serial ports and see section 2, pg. 232)
(f) a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, call etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1, which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Furthermore Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object. (As shown diagrammatically in FIG. 1, the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)") **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **Cohen** as described above.

U.S. Patent No. 5,455,854	Prior Art
(g) means for controlling the telephony element by the object oriented operating system utilizing the logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object. The "means for controlling" described in the '854 patent for carrying out the recited telephony functions relates to a computer system (See pg. 5 of Response dated June 20, 1995) that controls a telephony element such as a handset 400 in an object-oriented operating system and a class such as TTelephoneLineHandle "permits selection of a … line via a telephone line configuration data object (1600)." (See col. 16, lns. 5-8) The "Select Line" data object 1600 has logic such as member functions to interface the handset 400 to the "central processing unit 100, such as a … microprocessor" of the computer 412 (see col. 7, lns. 38-43 and col. 11, lns. 39-55) by utilizing the "member functions for initiating telephone call connections (1602), determining the hook status of the line (1604), matching telephone feature types via type negotiation (1606)". See col. 16, lns. 1-16 & FIGS. 1, 4 & 16	**Sixtensson** teaches implementation of the claimed means for controlling the telephony element that are the same as or equivalent to the implementations described in the specification of the '854 patent. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1; also section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A." Storage of "offHook" is also shown diagrammatically in FIG. 1.)
Claim 2	
The apparatus as recited in claim 1, including means for translating information received from the telephony element into information the object oriented operating system can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the means for translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.")

U.S. Patent No. 5,455,854	Prior Art
Claim 3	
The apparatus as recited in claim 1, including means for translating information received from the telephony object into information the telephony element can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the claimed means for translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice)
Claim 4	
The apparatus as recited in claim 1, including means for attaching the telephony element to the processor. The means described in the '854 patent relates to a telephone line 502 that attaches the handset 500 to the processor of computer 508. (See FIG. 5)	**Sixtensson** teaches a means for attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM.)
Claim 5	
The apparatus as recited in claim 4, including means for connecting a telephone line to the processor. The means described in the '854 patent relates to a jack 506 for connecting telephone line 504 to computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)
Claim 6	
The apparatus as recited in claim 4, including means for connecting a handset to the processor. The means described in the '854 patent relates to a telephone line 502 for connecting handset 500 to the processor of the computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.)

U.S. Patent No. 5,455,854	Prior Art
Claim 7	
The apparatus as recited in claim 4, including means for setting up a call to the processor. The means described in the '854 patent relates to "objects [that] could represent … signals or procedures, including … call setup … call features." (Abstract)	**Sixtensson** teaches a means for setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM and FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor). (See also section 3, pg. 233)
Claim 8	
The apparatus as recited in claim 4, including means for passing information between the telephony element and the processor. The means described in the '854 patent relate to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (Col. 12, lns. 29-33)	**Sixtensson** teaches a means for passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232)
Claim 10	
The apparatus as recited in claim 1, including means for enabling features of the telephony element via the telephony object. The means described in the '854 patent relates to a computer that enables classes of features of a handset via objects of the classes. (Col. 19, lns. 18-33)	**Sixtensson** teaches a means for enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables offhook (T1)).

U.S. Patent No. 5,455,854	Prior Art
Claim 11	
The apparatus as recited in claim 1, including means for servicing queries between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to a "computer [that] may … be able to control and/or query the handset" (Col. 11, lns. 50-55) and the computer has an object-oriented operating system.	**Sixtensson** teaches including means for servicing queries between a telephony element and the operating system (A person having ordinary skill in the art would understand that it is inherent or obvious for the SUN 3/50 workstation to service queries between a telephony element such as telephone T1 and an operating system) **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **Cohen.**
Claim 12	
The apparatus as recited in claim 1, including means for exchanging notification information between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to an "application 1100 [that] sets up to receive certain notifications regarding telephone systems. (Col. 14, lns. 22-25 & FIG. 11) The application 1100 resides on a computer that has an object-oriented operating system.	**Sixtensson** teaches a means for exchanging notification information between a telephony element and an operating system (it is inherent and would have been obvious for the SUN 3/50 workstation to have an application that receives notifications regarding telephone systems). **Furthermore,** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **Cohen**.
Claim 13	
A method for enabling telephony elements on a computer system, including a processor with an attached storage, display and telephony element, comprising the steps of:	**Sixtensson**, teaches the claimed method for enabling telephony elements on a computer system (section 1, pg. 232: "the prototyping system [2] … used … when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." See FIG. 1 **Sixtensson** teaches a processor with an attached storage (section 2, pg. 232: "LIM is a micro processor controlled unit" and FIG. 1: shows that LIM includes devices which may storage signals such as LPU, IOU, BSU, ELU). **Sixtensson** teaches a display (FIG. 1: shows that the SUN 3/50 workstation includes a display) **Sixtensson** teaches a telephony element (FIG. 1: telephone T1)

U.S. Patent No. 5,455,854	Prior Art
(a) controlling operations of the processor with an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage;	**Cohen** teaches an object-oriented operating system including objects (pgs. 201 & 202) **Cohen** teaches objects containing logic and data (pg. 202: "[t]he code invoked by calling a 'method' (a.k.a. procedure)" & "the binding of methods to code is implemented through attributes"). **Cohen** teaches an object oriented programming environment that extends software features for files and directories (pgs. 202 & 209). **APA of the '854 patent** discloses known benefits of object oriented applications for communications (col. 6, lns. 14-20: frameworks are sets of object classes that collaborate to execute computing responsibilities; col. 6, lines 23-27 & col. 7, lns. 3-6: "developers … extend the framework" thereby extending the software application "and create customized solutions in a particular area of expertise" such as "frameworks that provide basic system software services such as **communications**." (emphasis added). These frameworks provide operating system level functions. **Sixtensson** teaches that it is beneficial to use an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232) & the **APA** discloses the benefits of object oriented frameworks that extend software services for communications. As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings of **Cohen**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 and section 6, pg. 237) **Cohen** discloses supporting inheritance (pgs. 202 & 213: "the LINK attribute, provid[es] the same masquerading as the multiple inheritance model."). **APA of the '854 patent** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lns. 34-37) **Cohen** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (pg. 203: "to implement version selection … the properties are represented in data structures … laid out in the memory allocated to the … object").

U.S. Patent No. 5,455,854	Prior Art
(b) creating a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the step of creating the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1: which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object (FIG. 1: shows the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)". **Moreover** it is obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **Cohen** as described above.

U.S. Patent No. 5,455,854	Prior Art
(c) controlling the telephony element by the object-oriented operating system utilizing logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object.	**Sixtensson** teaches the claimed step of controlling the telephony element. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1; also section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A." Storage of "offHook" is also shown diagrammatically in FIG. 1.)
Claim 14	
The method as recited in claim 13, including the step of translating information received from the telephony element into information the object oriented operating system can utilize.	**Sixtensson** teaches the step of translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.")

U.S. Patent No. 5,455,854	Prior Art
Claim 15	
The method as recited in claim 13, including the step of translating information received from the telephony object into information the telephony element can utilize.	**Sixtensson** teaches the step of translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and would have been obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice.)
Claim 16	
The method as recited in claim 13, including the step of attaching the telephony element to the processor.	**Sixtensson** teaches the step of attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM)
Claim 17	
The method as recited in claim 16, including the step of connecting a telephone line to the processor.	**Sixtensson** teaches the step of connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)
Claim 18	
The method as recited in claim 16, including the step of connecting a handset to the processor.	**Sixtensson** teaches the step of connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.)
Claim 19	
The method as recited in claim 16, including the step of setting up a call to the processor.	**Sixtensson** teaches the step of setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM. FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor. See also section 3, pg. 233)

U.S. Patent No. 5,455,854	Prior Art
Claim 20	
The method as recited in claim 16, including the step of passing information between the telephony element and the processor.	**Sixtensson** teaches a step of passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232).
Claim 22	
The method as recited in claim 13, including the step of enabling features of the telephony element via the telephony object.	**Sixtensson** teaches the step of enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables "offhook (T1)").
Claim 23	
The method as recited in claim 13, including the step of exchanging status information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging status information between a telephony element and an operating system (FIG. 1: a telephony element such as telephone T1 exchanges status information indicating that it is offhook (e.g., "offhook (T1)") to the SUN 3/50 workstation which has an operating system) Furthermore, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **Cohen**.
Claim 24	
The method as recited in claim 13, including the step of exchanging notification information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging notification information between a telephony element and an operating system (section 3, pg. 233 & FIG. 2: information for ordering a call transfer is exchanged between a telephone element such as telephone T1 used by a user and the system, which inherently has an operating system). Moreover, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **Cohen**.

The Sixtensson article discloses all of the elements of claim 1 of the '854 patent implemented on a telephony apparatus except for the final computer code including an object-oriented operating system and object-oriented programming language. Sixtensson discloses an overall object-oriented method for automating and extending the Plain Ordinary Telephony Service (POTS) with new telecommunication service features. In this regard,

Sixtensson touts the benefits of using object-oriented approaches in achieving goals of extensibility and reuse in telephony.

In developing the complete software design, Sixtensson discloses that objects are mapped to processes and procedures in Object-oriented SDL (OSDL). (pg. 234) However, Sixtensson explains that the processes and procedures in OSDL are ultimately mapped to an Ada implementation language. As such, Sixtensson substantially generates the complete software design in an object-oriented manner leaving only a choice of an operating system and programming language.

The APA of the '854 patent discloses that it is beneficial to use object-oriented approaches for extending communications service features. For instance, the APA discloses that frameworks are sets of objects that collaborate to execute defined sets of computing responsibilities and describes that frameworks allow developers to extend the framework and software applications as well as create customized solutions in a particular area of expertise. (Col. 6, lns. 14-16, 27-29 & 43-44) The APA discloses that one such area of expertise relates to frameworks "that provide system software services such as communications." (Col. 7, lns. 3-6). As such, the APA discloses that it was well known to use object-oriented approaches for extending communications service features which are operating system level functions.

The Cohen article discloses a programming support environment which utilizes objects for extending software service features related to files and directories. (pgs. 202 & 209) Additionally, Cohen touts the benefits of using object-oriented approaches in achieving goals of extensibility for file systems and directories. (See *id*.)

The Cohen article enables the object-oriented benefits adopted by Sixtensson to be fully realized by providing both an object-oriented operating system and a programming

support environment that utilizes objects. One skilled in the art would have found a clear motivation to modify the Sixtensson article with the teachings of the Cohen article, which clearly discloses extension of features in an object-oriented operating system that includes objects, thereby allowing the benefits of an object-oriented environment in telephony to be realized.

More particularly, one skilled in the art would have found a clear motivation to modify the Sixtensson article with the teachings of the Cohen article given that the APA discloses the benefits of utilizing object-oriented approaches at an operating system level such as communications. Additionally, as described above, both Sixtensson and Cohen disclose extension of software features utilizing object-oriented approaches. In view of the APA's disclosure of applying object-oriented techniques to extend communications service features, a skilled artisan would have been motivated to modify the Sixtensson article with the teachings of the Cohen article in order to develop a finalized system design implementation, for extending telephony features, that utilizes objects as opposed to translating objects to an Ada implementation language.

The result of applying such teachings from the Cohen article to the Sixtensson article was predictable, namely, on using a programming support environment that utilizes objects to generate a complete or finalized system design implementation as opposed to translating objects to Ada, Sixtensson's automated telephony interface would have provided the same functionality while also benefitting from the advantages of an object-oriented operating system. Therefore, it would have been obvious to combine these two references to obtain the invention claimed in claim 1.

Independent claim 13 is a method version of claim 1, and reads on the combination of the Sixtensson article in view of the Cohen article according to the same analysis applied above in connection with claim 1. Dependent claims 2-8, 10-12, 14-20 and 22-24 would have been obvious for the reasons stated in the above claims chart for the first ground of rejection.

(b) ***Second Ground for Rejection***: Claims 1-8, 10-20 and 22-24 would have been obvious over the Sixtensson article in view of NeXTSTEP I, NeXTSTEP II and NeXTSTEP III.

By comparing the content of the Sixtensson article, NeXTSTEP I, NeXTSTEP II and NeXTSTEP III to claims 1-8, 10-20, and 22-24 of the '854 patent, it will become clear to one of ordinary skill in the art that the claimed subject matter would have been obvious.

U.S. Patent No. 5,455,854	**Prior Art**
Claim 1	
A telephony apparatus, comprising:	**Sixtensson**, discloses a telephony apparatus (section 1, pg. 232: "the prototyping system [2] has been used repeatedly when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." See also FIG. 1
(a) a processor;	**Sixtensson** teaches a processor (section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) … LIM is a micro processor controlled unit.")
(b) a storage attached to and controlled by the processor;	**Sixtensson** discloses a storage attached to and controlled by the processor (FIG. 1 and section 2, pg. 232: A person having ordinary skill in the art would understand that a SUN 3/50 workstation includes a storage such as for example a memory and that the LIM which "is a micro processor controlled unit" may control the storage of the SUN 3/50 workstation by accessing the "serial ports".)

U.S. Patent No. 5,455,854	Prior Art
(c) an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage and controlling operations of the processor;	**NeXTSTEP I** teaches an object-oriented operating system and object-oriented applications written in the same programming language (pg. xii: "This book is about the third component of the development environment–the programming language. All NeXTSTEP software kits are written in the Objective C language. To get the benefit of the kits, applications must also use Objective C." **APA of the '854 patent** teaches that a NeXTStep App Kit is part of an operating system (col. 7) and the quote above on pg. xii of NeXTSTEP discloses that users in the prior art developed systems using the same NextStep system for an object-oriented operating system as well as object-oriented applications.) **NeXTSTEP II** discloses benefits of object-oriented applications for telephony (pgs. 13-1 – 13-4) **NeXTSTEP III** discloses benefits of object-oriented applications for telephony (pgs. 13-3 & 13-7). **Sixtensson** teaches that it is beneficial to use an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232). As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings **NeXTSTEP I, II** & **III**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 and section 4, pg. 235) **NeXTSTEP I** teaches supporting encapsulation, polymorphism, and inheritance (pgs. 12-18) **APA of the '854 patent** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lines 34-37) **NeXTSTEP I** discloses objects containing logic and data (pgs. 1 & 5)

U.S. Patent No. 5,455,854	Prior Art
(d) a display attached to the processor under the control of the object oriented operating system;	**Sixtensson** teaches a display attached to the processor under control of an operating system (FIG. 1: shows that the SUN 3/50 workstation includes a display attached to the LIM via "serial ports." The LIM is a micro processor. See also section 2, pg. 232) It would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II** & **III** as described above.
(e) a telephony element attached to the processor;	**Sixtensson** teaches a telephony element attached to the processor (FIG. 1: shows that telephones T1 and T2 are attached to LIM via serial ports and see section 2, pg. 232)
(f) a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, call etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1, which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Furthermore Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object. (As shown diagrammatically in FIG. 1, the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)") **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II** & **III** as described above.

U.S. Patent No. 5,455,854	Prior Art
(g) means for controlling the telephony element by the object oriented operating system utilizing the logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object. The "means for controlling" described in the '854 patent for carrying out the recited telephony functions relates to a computer system (See pg. 5 of Response dated June 20, 1995) that controls a telephony element such as a handset 400 in an object-oriented operating system and a class such as TTelephoneLineHandle "permits selection of a … line via a telephone line configuration data object (1600)." (See col. 16, lns. 5-8) The "Select Line" data object 1600 has logic such as member functions to interface the handset 400 to the "central processing unit 100, such as a … microprocessor" of the computer 412 (see col. 7, lns. 38-43 and col. 11, lns. 39-55) by utilizing the "member functions for initiating telephone call connections (1602), determining the hook status of the line (1604), matching telephone feature types via type negotiation (1606)". See col. 16, lns. 1-16 & FIGS. 1, 4 & 16	**Sixtensson** teaches implementation of the claimed means for controlling the telephony element that are the same as or equivalent to the implementations described in the specification of the '854 patent. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1; also section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A." Storage of "offHook" is also shown diagrammatically in FIG. 1.) **NeXTSTEP II** also discloses a classified NXPhoneCall object that has member functions to: (1) Initiate a call connection (pg. 13-2:"-(void)pickUp" takes the phone off-hook to initiate an outgoing call); (2) monitor call progress (pg. 13-2: "(void)remoteBusy" responds to busy signal); and (3) activate call features (pg. 13-2: "(void)pickUp" answers an incoming call). See also pgs. 13-7 – 13-8 of **NeXTSTEP III**.

U.S. Patent No. 5,455,854	Prior Art
Claim 2	
The apparatus as recited in claim 1, including means for translating information received from the telephony element into information the object oriented operating system can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the means for translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.") **NeXTSTEP III** discloses the means for translating information (pg. 13-6: "the Phone Server is able to translate information received over the phone line into a form that's useful for applications")
Claim 3	
The apparatus as recited in claim 1, including means for translating information received from the telephony object into information the telephony element can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the claimed means for translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice) **NeXTSTEP III** discloses the means for translating information (pg. 13-6)
Claim 4	
The apparatus as recited in claim 1, including means for attaching the telephony element to the processor. The means described in the '854 patent relates to a telephone line 502 that attaches the handset 500 to the processor of computer 508. (See FIG. 5)	**Sixtensson** teaches a means for attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM.)
Claim 5	
The apparatus as recited in claim 4, including means for connecting a telephone line to the processor. The means described in the '854 patent relates to a jack 506 for connecting telephone line 504 to computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)

U.S. Patent No. 5,455,854	Prior Art
Claim 6	
The apparatus as recited in claim 4, including means for connecting a handset to the processor. The means described in the '854 patent relates to a telephone line 502 for connecting handset 500 to the processor of the computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.)
Claim 7	
The apparatus as recited in claim 4, including means for setting up a call to the processor. The means described in the '854 patent relates to "objects [that] could represent … signals or procedures, including … call setup … call features." (Abstract)	**Sixtensson** teaches a means for setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM and FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor). (See also section 3, pg. 233)
Claim 8	
The apparatus as recited in claim 4, including means for passing information between the telephony element and the processor. The means described in the '854 patent relate to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (Col. 12, lns. 29-33)	**Sixtensson** teaches a means for passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232)

U.S. Patent No. 5,455,854	Prior Art
Claim 10	
The apparatus as recited in claim 1, including means for enabling features of the telephony element via the telephony object. The means described in the '854 patent relates to a computer that enables classes of features of a handset via objects of the classes. (Col. 19, lns. 18-33)	**Sixtensson** teaches a means for enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables offhook (T1)). **NeXTSTEP II** discloses a means for enabling classes of features of a handset via objects of the classes (pgs. 13-1 – 13-4: relating to classes of objects for a phone kit) **NeXTSTEP III** discloses a means for enabling classes of features of a handset via objects of the classes (pgs. 13-7 – 13-13)
Claim 11	
The apparatus as recited in claim 1, including means for servicing queries between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to a "computer [that] may … be able to control and/or query the handset" (Col. 11, lns. 50-55) and the computer has an object-oriented operating system.	**Sixtensson** teaches including means for servicing queries between a telephony element and the operating system (A person having ordinary skill in the art would understand that it is inherent or obvious for the SUN 3/50 workstation to service queries between a telephony element such as telephone T1 and an operating system) **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II** & **III.** **NeXTSTEP III** discloses the means for servicing queries (pg. 13-6: "Communication between Kit and … Server is through … Objective C messages.")
Claim 12	
The apparatus as recited in claim 1, including means for exchanging notification information between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to an "application 1100 [that] sets up to receive certain notifications regarding telephone systems. (Col. 14, lns. 22-25 & FIG. 11) The application 1100 resides on a computer that has an object-oriented operating system.	**Sixtensson** teaches a means for exchanging notification information between a telephony element and an operating system (it is inherent and would have been obvious for the SUN 3/50 workstation to have an application that receives notifications regarding telephone systems). **Furthermore,** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II** & **III**. **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to Phone Server and for receiving notifications from the Server of activity on the phone line …")

U.S. Patent No. 5,455,854	Prior Art
Claim 13	
A method for enabling telephony elements on a computer system, including a processor with an attached storage, display and telephony element, comprising the steps of:	**Sixtensson**, teaches the claimed method for enabling telephony elements on a computer system (section 1, pg. 232: "the prototyping system [2] … used … when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." See FIG. 1 **Sixtensson** teaches a processor with an attached storage (section 2, pg. 232: "LIM is a micro processor controlled unit" and FIG. 1: shows that LIM includes devices which may storage signals such as LPU, IOU, BSU, ELU). **Sixtensson** teaches a display (FIG. 1: shows that the SUN 3/50 workstation includes a display) **Sixtensson** teaches a telephony element (FIG. 1: telephone T1)

U.S. Patent No. 5,455,854	Prior Art
(a) controlling operations of the processor with an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage;	**NeXTSTEP I** teaches controlling operations of a processor with an object-oriented operating system and object-oriented applications written in the same programming language for telephony (pg. xii: "This book is about the third component of the development environment–the programming language. All NeXTSTEP software kits are written in the Objective C language. To get the benefit of the kits, applications must also use Objective C", pgs. 13-1 – 13-4) **APA of the '854 patent** teaches that a NeXTStep App Kit is part of an operating system (col. 7) and the quote above on pg. xii of **NeXTSTEP I** discloses that users in the prior art developed systems using the same NextStep system for an object-oriented operating system as well as object-oriented applications). **NeXTSTEP II** discloses benefits of object-oriented applications for telephony (pgs. 13-1 – 13-4) **NeXTSTEP III** discloses benefits of object-oriented applications for telephony (pgs. 13-3 & 13-7). **Sixtensson** teaches that it is beneficial to use an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232). As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings of **NeXTSTEP I, II** & **III**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 and section 6, pg. 237) **NeXTSTEP I** discloses supporting encapsulation, polymorphism and inheritance (pgs. 12- 18). **APA of the '854 patent** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lns. 34-37) **NeXTSTEP I** teaches objects containing logic and data (pgs. 1 & 5)

U.S. Patent No. 5,455,854	Prior Art
(b) creating a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the step of creating the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1: which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object (FIG. 1: shows the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)". **Moreover** it is obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II** & **III** as described above.

U.S. Patent No. 5,455,854	Prior Art
(c) controlling the telephony element by the object-oriented operating system utilizing logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object.	**Sixtensson** teaches the claimed step of controlling the telephony element. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1; also section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A." Storage of "offHook" is also shown diagrammatically in FIG. 1.) **NeXTSTEP II** also discloses a classified NXPhoneCall object that has member functions to: (1) Initiate a call connection (pg. 13-2:"-(void)pickUp" takes the phone off-hook to initiate an outgoing call); (2) monitor call progress (pg. 13-2: "(void)remoteBusy" responds to busy signal); and (3) activate call features (pg. 13-2: "(void)pickUp" answers an incoming call). See also pgs. 13-7 – 13-8 of **NeXTSTEP III**.

U.S. Patent No. 5,455,854	Prior Art
Claim 14	
The method as recited in claim 13, including the step of translating information received from the telephony element into information the object oriented operating system can utilize.	**Sixtensson** teaches the step of translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.") **NeXTSTEP III** discloses the step of translating information (pg. 13-6: "Phone Server … translate[s] information over the phone line into a form … used for applications").
Claim 15	
The method as recited in claim 13, including the step of translating information received from the telephony object into information the telephony element can utilize.	**Sixtensson** teaches the step of translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and would have been obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice.) **NeXTSTEP III** discloses the step of translating information (pg. 13-6).
Claim 16	
The method as recited in claim 13, including the step of attaching the telephony element to the processor.	**Sixtensson** teaches the step of attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM)
Claim 17	
The method as recited in claim 16, including the step of connecting a telephone line to the processor.	**Sixtensson** teaches the step of connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)
Claim 18	
The method as recited in claim 16, including the step of connecting a handset to the processor.	**Sixtensson** teaches the step of connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.)

U.S. Patent No. 5,455,854	Prior Art
Claim 19	
The method as recited in claim 16, including the step of setting up a call to the processor.	**Sixtensson** teaches the step of setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM. FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor. See also section 3, pg. 233)
Claim 20	
The method as recited in claim 16, including the step of passing information between the telephony element and the processor.	**Sixtensson** teaches a step of passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232). **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to Phone Server and for receiving notifications from the Server of activity on the phone line …")
Claim 22	
The method as recited in claim 13, including the step of enabling features of the telephony element via the telephony object.	**Sixtensson** teaches the step of enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables "offhook (T1)"). **NeXTSTEP II** discloses the step of enabling classes of features of a handset via objects of the classes (pgs. 13-1 – 13-4: relating to classes of objects for a phone kit) **NeXTSTEP III** discloses the step of enabling classes of features of a handset via objects of the classes (pgs. 13-7 – 13-13)

U.S. Patent No. 5,455,854	Prior Art
Claim 23	
The method as recited in claim 13, including the step of exchanging status information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging status information between a telephony element and an operating system (FIG. 1: a telephony element such as telephone T1 exchanges status information indicating that it is offhook (e.g., "offhook (T1)") to the SUN 3/50 workstation which has an operating system) Furthermore, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II** & **III**. **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to the Phone Server and for receiving notifications from the Server of activity on the phone line …")
Claim 24	
The method as recited in claim 13, including the step of exchanging notification information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging notification information between a telephony element and an operating system (section 3, pg. 233 & FIG. 2: information for ordering a call transfer is exchanged between a telephone element such as telephone T1 used by a user and the system, which inherently has an operating system). Moreover, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II** & **III**. **NeXTSTEP III** discloses the step of exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to Phone Server and for receiving notifications from the Server of activity on the phone line …")

The Sixtensson article discloses all of the elements of claim 1 of the '854 patent implemented on a telephony apparatus except for the final computer code including an object-oriented operating system and interface. As described above, Sixtensson teaches an overall object-oriented method for automating and extending the Plain Ordinary Telephony Service (POTS) with new service features. Moreover, as pointed out above, Sixtensson touts the benefits of using object-oriented approaches in attaining the goals of extensibility and reuse in telephony.

In generating the entire software design, Sixtensson first makes a conceptual model and analyzes it with System Interactive Diagrams (SIDs) to recognize objects and their operations. (pg. 233) The second step is to evaluate a library to determine whether existing objects can be reused. (pg. 233) If the objects do not exist, the Sixtensson article discloses that a decision is made regarding whether new objects should be put in the library to be classified as components and Object Interactive Diagrams (OIDs) are made to capture the behavior of the system and the operations of the objects appear as signals in the OIDs. (pg. 233) Sixtensson then explains that the objects are classified as servers and nonservers using the OIDs and are mapped to processes and procedures in Object-oriented SDL (OSDL). (pg. 234) The processes and procedures in OSDL are ultimately mapped to an Ada implementation language. (pgs. 234 & 237) In view of the foregoing, Sixtensson substantially generates the entire design in an object-oriented manner leaving only a choice of an operating system and programming language.

NeXTSTEP I, NeXTSTEP II and NeXTSTEP III allows the object-oriented benefits embraced by Sixtensson to be fully realized by providing both an object-oriented operating system and an object-oriented application programming environment. One skilled in the art would have found a clear motivation to modify the Sixtensson article with the teachings of NeXTSTEP I, NeXTSTEP II and NEXTSTEP III, which clearly allows the benefits of an object-oriented environment in telephony to be realized. The result of applying such teachings from NeXTSTEP I, NeXTSTEP II and NeXTSTEP III to the Sixtensson article was predictable, namely, on using an object-oriented programming language for all applications as opposed to translating objects to Ada, Sixtensson's automated telephony interface would have provided the same functions while benefitting from the advantages of

an object-oriented operating system and object-oriented applications in the same programming language. Therefore, it would have been obvious to combine these four references to obtain the invention claimed in claim 1.

Independent claim 13 is a method version of claim 1, and reads on the combination of the Sixtensson article in view of NeXTSTEP I, NeXTSTEP II and NeXTSTEP III according to the same analysis applied above in connection with claim 1. Dependent claims 2-8, 10-12, 14-20 and 22-24 would have been obvious for the reasons stated in the above claims chart for this second ground of rejection.

(c) ***Third Ground for Rejection***: Claims 1-24 would have been obvious over the Sixtensson article, NeXTSTEP I, NeXTSTEP II, NeXTSTEP III and NeXTSTEP IV in view of the Lotito patent.

By comparing the content of the Sixtensson article, NeXTSTEP I, NeXTSTEP II, NeXSTEP III, NeXTSTEP IV and the Lotito patent to claims 1-24 of the '854 patent, it will become clear that the claimed subject matter would have been obvious to one of ordinary skill in the art.

U.S. Patent No. 5,455,854	**Prior Art**
Claim 1	
A telephony apparatus, comprising:	**Sixtensson** discloses a telephony apparatus (section 1, pg. 232: "the prototyping system [2] has been used repeatedly when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." See also FIG. 1 and telephones T1 and T2. See also section 2, pg. 232 "Though the … work relied on the prototyping system, the method outlined … is of a general nature" and as such a skilled artisan wound understand that the software design could be used in a real world system.) **Lotito** discloses a telephony apparatus (col. 5, lns. 39-45: "an automated telephone voice service system 100." See also FIG. 1.)

U.S. Patent No. 5,455,854	Prior Art
(a) a processor;	**Sixtensson** teaches a processor (section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) … LIM is a micro processor controlled unit.") **Lotito** teaches a processor (col. 14, lns. 26-46: "A standard processor module 500 … a CPU 504". See also FIGS. 5, 4 & 2)
(b) a storage attached to and controlled by the processor;	**Sixtensson** discloses a storage attached to and controlled by the processor (FIG. 1 and section 2, pg. 232: A person having ordinary skill in the art would understand that a SUN 3/50 workstation includes a storage such as for example a memory and that the LIM which "is a micro processor controlled unit" may control the storage of the SUN 3/50 workstation by accessing the "serial ports".) **Lotito** teaches a storage attached to and controlled by the processor (col. 14, lns. 26-31: "A standard processor module 500 … including … a 4k X 16 RAM program memory". See also FIG. 5)

U.S. Patent No. 5,455,854	Prior Art
(c) an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage and controlling operations of the processor;	**NeXTSTEP I** teaches an object-oriented operating system and object-oriented applications written in the same programming language (pg. xii: "This book is about the third component of the development environment–the programming language. All NeXTSTEP software kits are written in the Objective C language. To get the benefit of the kits, applications must also use Objective C." **APA of the '854 patent** teaches that a NeXTStep App Kit is part of an operating system (col. 7) and the quote above on pg. xii of NeXTSTEP discloses that users in the prior art developed systems using the same NextStep system for an object-oriented operating system as well as object-oriented applications.) **NeXTSTEP II** discloses benefits of object-oriented applications for telephony (pgs. 13-1 – 13-4). **NeXTSTEP III** discloses benefits of object-oriented applications for telephony (pgs. 13-3 & 13-7). **NeXTSTEP IV** discloses benefits of object-oriented applications for a Display PostScript system (pg. 1) **Lotito** teaches an operating system used in telephony in which the applications and the operating system are programmed in the same programming language (col. 75, lns. 39-43, col. 94, lns. 47-51 & col. 23, lns. 13-14, 60-61) **Sixtensson** teaches benefits using an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232). As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings **NeXTSTEP I, II, III, IV** & **Lotito**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 & section 4, pg. 235) **NeXTSTEP I** teaches supporting encapsulation, polymorphism and inheritance (pgs. 12-18). **NeXTSTEP IV** teaches supporting encapsulation and inheritance (pgs. xxii, 2 & 10). **APA of the '854 patent** teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lines 34-37) **NeXTSTEP I** teaches objects containing logic and data (pgs. 1 & 5). **NeXTSTEP IV** teaches objects containing logic and data (pgs. 1 & 2).

U.S. Patent No. 5,455,854	Prior Art
(d) a display attached to the processor under the control of the object oriented operating system;	**Sixtensson** teaches a display attached to the processor under control of an operating system (FIG. 1: shows that the SUN 3/50 workstation includes a display attached to the LIM via "serial ports." The LIM is a micro processor. See also section 2, pg. 232) It would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II, III, IV** & **Lotito** as described above. **Lotito** discloses the claimed display (Col. 57, lines 49-68: a system terminal 270 has a "visual display unit (vdu)" & the terminal 270 "interact[s] directly with … processors").
(e) a telephony element attached to the processor;	**Sixtensson** teaches a telephony element attached to the processor (FIG. 1: shows that telephones T1 and T2 are attached to LIM via serial ports and see section 2, pg. 232) Lotito discloses a telephony element attached to a processor (FIGS. 2, 4 & 5-6, information processing system is attached to processor module 602 via real time subsystem 230)
(f) a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, call etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1: which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Furthermore Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object. (As shown in FIG. 1, the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)") **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II, III, IV** & **Lotito** as described above.

U.S. Patent No. 5,455,854	Prior Art
(g) means for controlling the telephony element by the object oriented operating system utilizing the logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object. The "means for controlling" described in the '854 patent for carrying out the recited telephony functions relates to a computer system (See pg. 5 of Response dated June 20, 1995) that controls a telephony element such as a handset 400 in an object-oriented operating system and a class such as TTelephoneLineHandle "permits selection of a … line via a telephone line configuration data object (1600)." (See col. 16, lns. 5-8) The "Select Line" data object 1600 has logic such as member functions to interface the handset 400 to the "central processing unit 100, such as a … microprocessor" of the computer 412 (see col. 7, lns. 38-43 and col. 11, lns. 39-55) by utilizing the "member functions for initiating telephone call connections (1602), determining the hook status of the line (1604), matching telephone feature types via type negotiation (1606)". See col. 16, lns. 1-16 & FIGS. 1, 4 & 16	**Sixtensson** teaches implementation of the claimed means for controlling the telephony element that are the same as or equivalent to the implementations described in the specification of the '854 patent. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1 and section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A" **NeXTSTEP II** also discloses a classified NXPhoneCall object that has member functions to: (1) Initiate a call connection (pg. 13-2:"-(void)pickUp" takes the phone off-hook to initiate an outgoing call); (2) Monitor call progress (pg. 13-2: "(void)remoteBusy" responds to busy signal); and (3) Activate call features (pg. 13-2: "(void)pickUp" answers an incoming call). See also pgs. 13-7 – 13.8 of **NeXTSTEP III**. **Lotito** discloses telephone system services to: (1) Interface to processor (FIGS. 2, 4-6: Processor 602 interfaced to information processing system 250); (2) Initiate a call connection (col. 3, lines 6-9: "A voice line communication coupling between the calling telephone and … message basket." See also col. 135, lines 53-54);

U.S. Patent No. 5,455,854	Prior Art
	(3) Monitor call progress (col. 135, lines 55-56); & (4) Activate call features (col. 121, lines 3-8).
Claim 2	
The apparatus as recited in claim 1, including means for translating information received from the telephony element into information the object oriented operating system can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the means for translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.") **NeXTSTEP III** discloses the means for translating information (pg. 13-6: "Phone Server is able to translate information received over the phone into a form that's useful for applications").
Claim 3	
The apparatus as recited in claim 1, including means for translating information received from the telephony object into information the telephony element can utilize. The means described in the '854 patent relates to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (See col. 12, lns. 23-33)	**Sixtensson** teaches the claimed means for translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice) **NeXTSTEP III** discloses the means for translating information (pg. 13-6).
Claim 4	
The apparatus as recited in claim 1, including means for attaching the telephony element to the processor. The means described in the '854 patent relates to a telephone line 502 that attaches the handset 500 to the processor of computer 508. (See FIG. 5)	**Sixtensson** teaches a means for attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM) **Lotito** teaches the means for attaching (FIGS. 2, 4, 6: buses 414 and 416 for attaching the information processing system 250 to the processor 602.
Claim 5	
The apparatus as recited in claim 4, including means for connecting a telephone line to the processor. The means described in the '854 patent relates to a jack 506 for connecting telephone line 504 to computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)

U.S. Patent No. 5,455,854	Prior Art
Claim 6	
The apparatus as recited in claim 4, including means for connecting a handset to the processor. The means described in the '854 patent relates to a telephone line 502 for connecting handset 500 to the processor of the computer 508. (See FIG. 5)	**Sixtensson** teaches a means for connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.) **Lotito** discloses the means for connecting (Col. 57, lines 61-68: System Programmer's Terminal … interact[s] directly with … processors").
Claim 7	
The apparatus as recited in claim 4, including means for setting up a call to the processor. The means described in the '854 patent relates to "objects [that] could represent … signals or procedures, including … call setup … call features." (Abstract)	**Sixtensson** teaches a means for setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM and FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor). (See also section 3, pg. 233)
Claim 8	
The apparatus as recited in claim 4, including means for passing information between the telephony element and the processor. The means described in the '854 patent relate to "interface elements [that] standardize the methods used for passing information from and to various of the telephony system elements. The interface elements are advantageously implemented using objects …" (Col. 12, lns. 29-33)	**Sixtensson** teaches a means for passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232) **Lotito** teaches a means for passing information (FIGS. 2, 4, 6: buses 414, 416 pass information between processor 602 and information processing system 250)

U.S. Patent No. 5,455,854	Prior Art
Claim 9	
The apparatus as recited in claim 8, including means for exchanging DTMF tones between the telephony element and the processor. The means described in the '854 patent relates to "telephony equipment 1200 sending out information via objects which can be used to manage telephony transactions… TelephoneDigitsNotification at 1206 permits clients to receive DTMF digits generated by the remote endpoint." (Col. 14, lns. 31-44 & FIG. 12)	**Sixtensson** teaches a means for exchanging tones between the telephony element and the processor (FIG. 2: shows that signals for exchanging dialToneOn and dialToneOff tones are exchanged between a telephone such as telephone T1 used by a user and the system which uses an LIM and section 2, pg. 232: the signals from the telephone units are sent directly to the LIM which is a processor.) **Lotito** teaches DTMF tones (col. 119, lines 37-41: tones or tone sequences such as DTMF digit sequences can be applied to a channel. (See also col. 11, lns. 6-9) **Moreover**, it would have been obvious to modify the tones of Sixtensson to be DTMF tones that are exchanged between telephone T1 and the LIM.
Claim 10	
The apparatus as recited in claim 1, including means for enabling features of the telephony element via the telephony object. The means described in the '854 patent relates to a computer that enables classes of features of a handset via objects of the classes. (Col. 19, lns. 18-33)	**Sixtensson** teaches a means for enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables offhook (T1)). **NeXTSTEP II** discloses a means for enabling classes of features of a handset via objects of the classes (pgs. 13-1 – 13-4: relating to classes of objects for a phone kit) **NeXTSTEP III** discloses a means for enabling classes of features of a handset via objects of the classes (pgs. 13-7 – 13-13)
Claim 11	
The apparatus as recited in claim 1, including means for servicing queries between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to a "computer [that] may … be able to control and/or query the handset" (Col. 11, lns. 50-55) and the computer has an object-oriented operating system.	**Sixtensson** teaches including means for servicing queries between a telephony element and the operating system (A person having ordinary skill in the art would understand that it is inherent or obvious for the SUN 3/50 workstation to service queries between a telephony element such as telephone T1 and an operating system) **Moreover** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II, III, IV** & **Lotito.** **NeXTSTEP III** discloses the means for servicing queries (pg. 13-6: "Communication between Kit and … Server is through … Objective C messages.")

U.S. Patent No. 5,455,854	Prior Art
Claim 12	
The apparatus as recited in claim 1, including means for exchanging notification information between a telephony element and the object-oriented operating system. The means described in the '854 patent relates to an "application 1100 [that] sets up to receive certain notifications regarding telephone systems. (Col. 14, lns. 22-25 & FIG. 11) The application 1100 resides on a computer that has an object-oriented operating system.	**Sixtensson** teaches a means for exchanging notification information between a telephony element and an operating system (it is inherent and would have been obvious for the SUN 3/50 workstation to have an application that receives notifications regarding telephone systems). **Furthermore,** it would have been obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II, III, IV** & **Lotito**. **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to Phone Server and for receiving notifications from the Server of activity on the phone line …")
Claim 13	
A method for enabling telephony elements on a computer system, including a processor with an attached storage, display and telephony element, comprising the steps of:	**Sixtensson**, teaches the claimed method for enabling telephony elements on a computer system (section 1, pg. 232: "the prototyping system [2] … used … when extending the [Plain Ordinary Telephony Service] (POTS) software" and section 2, pg. 232: "The prototyping system is a Line Interface Module (LIM) from the MD 110 PABX system by Ericsson and a SUN 3/50 workstation." **Sixtensson** teaches a processor with an attached storage (section 2, pg. 232: "LIM is a micro processor controlled unit" and FIG. 1: shows that LIM includes devices which may storage signals such as LPU, IOU, BSU, ELU). **Sixtensson** teaches a display (FIG. 1: shows that the SUN 3/50 workstation includes a display) **Sixtensson** teaches a telephony element (FIG. 1: telephone T1) **Lotito** discloses the claimed telephony elements (col. 14, lines 26-46, col. 57, lns. 48-68 & FIGS. 2, 4, 5 & 6).

U.S. Patent No. 5,455,854	Prior Art
(a) controlling operations of the processor with an object oriented operating system, supporting encapsulation, polymorphism and inheritance, including objects, each of the objects containing logic and data resident in the storage;	**NeXTSTEP I** teaches controlling operations of a processor with an object-oriented operating system and object-oriented applications written in the same programming language (pg. xii: "This book is about the third component of the development environment–the programming language. All NeXTSTEP software kits are written in the Objective C language. To get the benefit of the kits, applications must also use Objective C") **APA of the '854 patent** teaches that a NeXTStep App Kit is part of an operating system (col. 7) and the quote above on pg. xii of **NeXTSTEP I** discloses that users in the prior art developed systems using the same NextStep system for an object-oriented operating system as well as object-oriented applications). **NeXTSTEP II** discloses benefits of object-oriented applications for telephony (pgs. 13-1 – 13-4). **NeXTSTEP III** discloses benefits of object-oriented applications for telephony (pgs. 13-3 & 13-7). **NeXTSTEP IV** discloses benefits of object-oriented applications for a Display PostScript system (pg. 1). **Lotito** teaches an operating system in an overall system for automated telephony in which the applications and the operating system are programmed in the same programming language (col. 75, lns. 39-43, col. 94, lns. 47-51 and col. 23, lns. 13-14, 60-61). **Sixtensson** teaches that it is beneficial to use an object-oriented approach with telephony (Abstract & section 1 on pgs. 231 and 232). As such, it would have been obvious to use an object-oriented operating system and object-oriented applications in **Sixtensson** in view of the teachings of **NeXTSTEP I, II, III, IV & Lotito**. **Sixtensson** teaches supporting encapsulation, including objects (section 4, pg. 235) **Sixtensson** teaches supporting polymorphism including objects (section 5, pg. 236: "There are many versions of source code describing object telephone. They are all interchangeable." A person with ordinary skill in the art would understand that the foregoing relates to supporting polymorphism including objects) **Sixtensson** teaches supporting inheritance, including objects (section 3, pg. 234 and section 6, pg. 237) **NeXTSTEP I** teaches supporting encapsulation, polymorphism and inheritance (pgs. 12 - 18). **NeXTSTEP IV** teaches supporting encapsulation and inheritance (pgs. xxii, 2 & 10).

U.S. Patent No. 5,455,854	Prior Art
	APA of the '854 patent teaches each of the objects containing logic and data resident in the storage and controlling operations of the processor (col. 5, lns. 34-37) **NeXTSTEP I** teaches objects containing logic and data (pgs. 1 & 5). **NeXTSTEP IV** teaches objects containing logic and data (pgs. 1 & 2).
(b) creating a telephony object, including logic for interfacing the telephony element to the processor and data for storing status information associated with the telephony element in the telephony object, and representative of the telephony element under the control of the object-oriented operating system, stored in the storage and displayed on the display; and	**Sixtensson** teaches the step of creating the claimed telephony object (section 3, pg. 233: "In fig 3, the vertical lines denote objects decomposed from the system, such as telephone A, toneDevice, switch, etc." and see section 4, pgs. 234 and 235. See also section 2, pg. 232 & FIG. 1: which shows that the telephony object "offhook" includes operations for interfacing the telephony element telephone T1 to the LIM which is a micro processor controlled unit and shows that the telephony object "offhook" is displayed on a display of the SUN 3/50 workstation) **Sixtensson** teaches that the data of the telephony object stores status information associated with the telephony element in the telephony object (FIG. 1: shows the telephony object stores status information associated with telephony element telephone T1, for example "offhook (T1)". **Moreover** it is obvious to modify the operating system of **Sixtensson** to be object-oriented in view of **NeXTSTEP I, II, III, IV** & **Lotito** as described above.

U.S. Patent No. 5,455,854	Prior Art
(c) controlling the telephony element by the object-oriented operating system utilizing logic in the telephony object to interface the telephony element to the processor by initiating a call connection, monitoring call progress, activating call features and storing status information in the data of the telephony object.	**Sixtensson** teaches the step of controlling the telephony element. (a) Interface to processor: (FIG. 1: the SUN 3/50 workstation controls the telephony element telephone T1; also section 3, pg. 233: "new objects should … be classified as components" and "the objects telephoneA, toneDevice, switch, call" have operations to interface the telephony element such as telephone T1 to the LIM, which is a micro processor) (b) Initiating a call connection: (section 4, pg. 234: discloses that a classified object may permit selection of a line via a data object and describes that "typical objects are … lines … and operations are connect two telephones" which teaches initiating a call connection) (c) monitoring call progress: (section 4, pg. 234: "typical objects are … lines" and "operations are … generate a busy tone" which teaches monitoring call progress) (d) activating call features: (FIG. 3 & section 4, pg. 234: "typical objects are … lines" and "operations are … send digits", for a call transfer (e.g., setCallTransfer) which teaches activating call features) (e) storing status information: (FIG. 3: when a telephony element such as telephone T1 "starts dialing" at sequence event 1, "offHook" status information is stored in the data of the telephony object such as "telephone subscriber A." Storage of "offHook" is also shown diagrammatically in FIG. 1.) **NeXTSTEP II** also discloses a classified NXPhoneCall object that has member functions to: (1) Initiate a call connection (pg. 13-2:"-(void)pickUp" takes the phone off-hook to initiate an outgoing call); (2) monitor call progress (pg. 13-2: "(void)remoteBusy" responds to busy signal); and (3) activate call features (pg. 13-2: "(void)pickUp" answers an incoming call). See also pgs. 13-7 – 13-8 of **NeXTSTEP III**. **Lotito** discloses telephone system services to: (1) Interface to processor (FIGS. 2, 4-6: Processor 602 interfaced to information processing system 250); (2) Initiate a call connection (col. 3, lines 6-9: "A voice line communication coupling between the calling telephone and … message basket." See also col. 135, lines 53-54);

U.S. Patent No. 5,455,854	Prior Art
	(3) Monitor call progress (col. 135, lines 55-56); & (4) Activate call features (col. 121, lines 3-8).
Claim 14	
The method as recited in claim 13, including the step of translating information received from the telephony element into information the object oriented operating system can utilize.	**Sixtensson** teaches the step of translating information received from the telephony element (FIG. 3 & section 3, pg. 233: "The directed lines among objects represent signal passing between objects.") **NeXTSTEP III** discloses the step of translating information (pg. 13-6: "Phone Server … translate[s] information received over the phone line into a form that's useful for applications").
Claim 15	
The method as recited in claim 13, including the step of translating information received from the telephony object into information the telephony element can utilize.	**Sixtensson** teaches the step of translating information received from the telephony object (section 3, pg. 233: "The directed lines among objects represent signal passing between objects" and as shown in FIG. 3 at least one of the signals (e.g., dialToneOn at sequence event 2) is received from a telephony object (e.g., dtsDatatoneDevice) by a telephony element such as telephone T1 that "start[ed] dialing". It is inherent and would have been obvious that the telephone unit T1 can utilize information received from the dtsDatatoneDevice.) **NeXTSTEP III** discloses the step of translating information (pg. 13-6).
Claim 16	
The method as recited in claim 13, including the step of attaching the telephony element to the processor.	**Sixtensson** teaches the step of attaching the telephony element to the processor (section 2, pg. 232: the "LIM is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines. It can work as an autonomous exchange as well as part of a system comprising several LIM's connected via a switch" and FIG. 1 which shows telephone lines attaching telephone T1 and telephone T2 to the LIM) **Lotito** teaches the step of attaching (FIGS. 2, 4, 6: buses 414 and 416 for attaching the information processing system 250 to the processor 602.
Claim 17	
The method as recited in claim 16, including the step of connecting a telephone line to the processor.	**Sixtensson** teaches the step of connecting a telephone line to a processor (section 2, pg. 232 & FIG. 1: "The signals from the telephones units [T1 and T2] are sent directly to the LIM … via serial ports".)

U.S. Patent No. 5,455,854	Prior Art
Claim 18	
The method as recited in claim 16, including the step of connecting a handset to the processor.	**Sixtensson** teaches the step of connecting a handset to the processor (section 2, pg. 232: The "LIM …. is a micro processor … and can connect a number of subscriber extension lines, trunk lines and operator lines" & FIG. 1 shows that a telephone line connects telephone T1 to the LIM which is a micro processor.) **Lotito** discloses the step of connecting (Col. 57, lines 61-68)
Claim 19	
The method as recited in claim 16, including the step of setting up a call to the processor.	**Sixtensson** teaches the step of setting up a call to the processor (section 4, pg. 234: "typical objects are … lines" and FIG. 1 shows that lines connected between the telephones T1 and T2 are used to setup a call to the LIM. FIG. 3 section 3, pg. 233: "the vertical lines denote objects … such as call" and "transfer Data". When a telephone unit starts "dialing a sequence *3*" an object such as telephone subscriber A sets up a call transfer ("setCallTransfer") to the LIM which is a processor. See also section 3, pg. 233)
Claim 20	
The method as recited in claim 16, including the step of passing information between the telephony element and the processor.	**Sixtensson** teaches a step of passing information between the telephony element and the processor (FIG. 3 and section 3, pg. 233: "the directed lines among objects represent signal passing between objects" and these signals may be passed between the telephone T1 and the LIM which is a micro processor. See FIG. 1 and section 2, pg. 232) **Lotito** teaches the step of passing information (FIGS. 2, 4, 6: buses 414, 416 pass information between processor 602 and information processing system 250). **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to Phone Server and for receiving notifications from the Server of activity on the phone line …")

U.S. Patent No. 5,455,854	Prior Art
Claim 21	
The method as recited in claim 20, including the step of exchanging DTMF tones between the telephony element and the processor.	**Sixtensson** teaches the step of exchanging tones between the telephony element and the processor (FIG. 2: shows that signals for exchanging dialToneOn and dialToneOff tones are exchanged between a telephone such as telephone T1 used by a user and the system which uses an LIM and section 2, pg. 232: the signals from the telephone units are sent directly to the LIM which is a processor.) **Lotito** teaches DTMF tones (col. 119, lines 37-41: tones or tone sequences such as DTMF digit sequences can be applied to a channel. (See also col. 11, lns. 6-9) **Moreover**, it would have been obvious to modify the tones of Sixtensson to be DTMF tones that are exchanged between telephone T1 and the LIM.
Claim 22	
The method as recited in claim 13, including the step of enabling features of the telephony element via the telephony object.	**Sixtensson** teaches the step of enabling features of the telephony element via the telephony object (section 3, pg. 233: "new objects … in the library [are] classified as components". It is inherent or would have been obvious for the SUN 3/50 workstation to enable the features associated with the classified objects of the telephone T1 via one of the objects and as shown in FIG. 1, the SUN 3/50 workstation enables "offhook (T1)"). **NeXTSTEP II** discloses the step of enabling classes of features of a handset via objects of the classes (pgs. 13-1 – 13-4: relating to classes of objects for a phone kit) **NeXTSTEP III** discloses the step of enabling classes of features of a handset via objects of the classes (pgs. 13-7 – 13-13)
Claim 23	
The method as recited in claim 13, including the step of exchanging status information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging status information between a telephony element and an operating system (FIG. 1: a telephony element such as telephone T1 exchanges status information indicating that it is offhook (e.g., "offhook (T1)") to the SUN 3/50 workstation which has an operating system) **NeXTSTEP III** discloses a means for exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to the Phone Server and for receiving notifications from the Server of activity on the phone line …") Furthermore, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II, III, IV** & **Lotito**.

U.S. Patent No. 5,455,854	Prior Art
Claim 24	
The method as recited in claim 13, including the step of exchanging notification information between a telephony element and the object-oriented operating system.	**Sixtensson** teaches the step of exchanging notification information between a telephony element and an operating system (section 3, pg. 233 & FIG. 2: information for ordering a call transfer is exchanged between a telephone element such as telephone T1 used by a user and the system, which inherently has an operating system). Moreover, it would have been obvious for the operating system of **Sixtensson** to be an object-oriented operating system in view of **NeXTSTEP I, II, III, IV** & **Lotito**. **NeXTSTEP III** discloses the step of exchanging notification information (pg. 13-6: "The Phone Kit … deliver[s] instructions to the Phone Server and for receiving notifications from the Server of activity on the phone line …")

The Sixtensson article discloses all of the elements of claim 1 of the '854 patent implemented on a telephony apparatus except for the final computer code including an object-oriented operating system and interface.

As described above, the Sixtensson article generates substantially the entire software design described therein in an object-oriented manner leaving only a choice of an operating system and programming language. NeXTSTEP I, NeXTSTEP II, NeXTSTEP III and NeXTSTEP IV allow the object-oriented benefits embraced by Sixtensson to be fully realized by providing both an object-oriented operating system and an object-oriented application programming environment. Furthermore, NeXTSTEP provided kits for telephony interfaces.

Moreover, Lotito teaches an operating system in an overall system for automated telephony in which the applications and the operating system are programmed in the same programming language, namely Pascal. Although the Lotito patent does not explicitly disclose an object-oriented operating system, a skilled artisan would understand that Lotito does teach applications and an operating system programmed in the same programming

language. In this regard, Lotito teaches usage of applications and an operating system that are designed in the same programming language. Lotitio's teachings thus provide motivation for a skilled artisan to modify Sixtensson such that the final computer code is in the same programming language without requiring translation of objects into an Ada implementation language.

As such, one of ordinary skill in the art would have found a clear motivation in the Lotito patent, which clearly addresses utilizing the same programming language for applications and an operating system in telephony, to modify the Sixtensson article with the teachings of NeXTSTEP I, NeXTSTEP II, NeXTSTEP III and NeXTSTEP IV to use an integrated Conceptual-C programming environment for the operating system and applications needed to implement Sixtensson's object-oriented system design. The result of applying such teachings from NeXTSTEP I, NeXTSTEP II, NeXTSTEP III and NeXTSTEP IV to the Sixtensson article in light of Lotito was predictable, namely, on using an object-oriented programming language for all applications as opposed to translating objects to Ada, Sixtensson's automated telephony interface would have provided the same functions while benefitting from the advantages of an object-oriented operating system and object-oriented applications in the same programming language. Therefore, it would have been obvious to combine these six references to obtain the invention claimed in claim 1.

Independent claim 13 is a method version of claim 1, and reads on the combination of the Sixtensson article, NeXTSTEP I, NeXTSTEP II, NeXTSTEP III and NeXTSTEP IV in view of the Lotito patent according the same analysis applied above in connection with claim 1. Dependent claims 2-12, and 14-24 would have been obvious for the reasons stated in the above claims chart for this third ground of rejection.

III. Conclusion

The Sixtensson article, the Cohen article, NeXTSTEP I, NeXTSTEP II, NeXTSTEP III, NeXTSTEP IV and the Lotito patent were not previously considered, and they are not cumulative with the references previously considered. Consideration of obvious combinations of these references in accordance with the proposed Grounds for Rejection leads to the conclusion that these references create substantial new questions of patentability for claims 1-24 of the '854 patent. The Requester further submits that claims 1-24 must be rejected as unpatentable.

Respectfully submitted,

/Scott E. Brient/

Scott E. Brient

Reg. No. 44,561

Customer No. 93485
Brient Intellectual Property Law, LLC
2300 Lakeview Parkway, Suite 700
Alpharetta, GA 30009
Phone: (678) 916-3828
Fax: (678) 835-9310

ELECTRONICALLY FILED USING THE EFS-WEB ELECTRONIC FILING SYSTEM OF THE UNITED STATES PATENT & TRADEMARK OFFICE ON April 28, 2010.

EXHIBIT 7

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/010,965	04/28/2010	5455854	0919/01030	8066

64967 7590 07/14/2010

LAW OFFICES OF PAUL E. KUDIRKA
40 BROAD STREET
SUITE 300
BOSTON, MA 02109

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 07/14/2010

Please find below and/or attached an Office communication concerning this application or proceeding.

Order Granting / Denying Request For Ex Parte Reexamination	**Control No.** 90/010,965	**Patent Under Reexamination** 5455854	
	Examiner Deandra M. Hughes	**Art Unit** 3992	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

The request for *ex parte* reexamination filed *28 April 2010* has been considered and a determination has been made. An identification of the claims, the references relied upon, and the rationale supporting the determination are attached.

Attachments: a)☐ PTO-892, b)☒ PTO/SB/08, c)☐ Other: _____

1. ☒ The request for *ex parte* reexamination is GRANTED.

RESPONSE TIMES ARE SET AS FOLLOWS:

For Patent Owner's Statement (Optional): TWO MONTHS from the mailing date of this communication (37 CFR 1.530 (b)). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).**

For Requester's Reply (optional): TWO MONTHS from the **date of service** of any timely filed Patent Owner's Statement (37 CFR 1.535). **NO EXTENSION OF THIS TIME PERIOD IS PERMITTED.** If Patent Owner does not file a timely statement under 37 CFR 1.530(b), then no reply by requester is permitted.

2. ☐ The request for *ex parte* reexamination is DENIED.

This decision is not appealable (35 U.S.C. 303(c)). Requester may seek review by petition to the Commissioner under 37 CFR 1.181 within ONE MONTH from the mailing date of this communication (37 CFR 1.515(c)). **EXTENSION OF TIME TO FILE SUCH A PETITION UNDER 37 CFR 1.181 ARE AVAILABLE ONLY BY PETITION TO SUSPEND OR WAIVE THE REGULATIONS UNDER 37 CFR 1.183.**

In due course, a refund under 37 CFR 1.26 (c) will be made to requester:

a) ☐ by Treasury check or,

b) ☐ by credit to Deposit Account No. _____, or

c) ☐ by credit to a credit card account, unless otherwise notified (35 U.S.C. 303(c)).

cc:Requester (if third party requester)

ORDER GRANTING REQUEST FOR *EX PARTE* REEXAMINATION

1. Substantial new questions of patentability ("**SNQ**") affecting **claims 1-24** of **USP 5,455,854** ("**'854 patent**") have been proposed by the third party requester ("**3PR**") in the *ex parte* reexamination request filed April, 28, 2010 ("**Request**").

References Cited in this Action

2. Sixtensson et al. "Reuse in the Telecommunication Domain Using Object Oriented Technology and ADA". Washington ADA Symposium Proceeding. June 1990. pgs. 231-239. ("**Sixtensson**")

3. Cohen et al. "Version Management in Gypsy". Siemens Research and Technology Laboratories. 1988. ("**Cohen**")

4. "NeXTSTEP™ Object-Oriented Programming and the Objective Language". NeXTSTEP Developer's Library Release 3. Addison-Wesley Publishing Company. April 1993. ("**NextStep**")

5. "Programming the Display PostScript ® System with NeXTSTEP™". Addison-Wesley Publishing Company. April 1993. ("**PostScript**")

6. USP 4,625,081 to Lotito published Nov. 26, 1986. ("**Lotito**")

Prosecution History

7. The prosecution history of the application (08/108,877) which became the **'854 patent** is presented below.

- On August 26, 1993, **claims 1-26** were presented for examination.
- On October 11, 1994, **claims 1-26** were rejected as clearly anticipated by any one of Babson, Hayden, Britton, Ljunblom, or Dickman.
- On January 11, 1995, independent **claims 1 and 14** were amended and **claims 4 and 17** were cancelled.
- On January 25, 1995, an Examiner interview was conducted wherein the breadth of **claim 1** with respect to the terms 'object oriented' and 'telephony objects' was discussed.

- On April 12, 1995, **claims 1-3, 5-16, 18-26** were finally rejected as being obvious over Staas in view of Rankin.
- On June 22, 1995, a second Examiner interview was conducted wherein the Examiner suggested claim amendments to distinguish the claimed object oriented operating system of independent **claims 1 and 14** from the prior art object oriented programming disclosures.
- On June 27, 1995, the Examiner issued a notice of allowance with Examiner's amendments, which were discussed in the interview of June 22, 1995. These amendments were as follows.

  In **claims 1 and 14**, the word "system" was deleted and the phrase "system supporting encapsulation, polymorphism, and inheritance" was inserted to put **claims 1-3, 5-16, 18-26** in condition for allowance.

8. The Examiner considers a teaching as to "*an object oriented operating system supporting encapsulation, polymorphism, and inheritance*" to form the proper basis of an SNQ for **claims 1-24** of the **'854 patent** because the Examiner of application 08/108,877 determined that the insertion of this phrase made the claims allowable over the prior art.

Decision

9. In the Request, **3PR** states that the following prior art references disclose the following features that form the basis of the SNQ:

- **NextStep** discloses supporting encapsulation, polymorphism, and inheritance on pages 12-18. *(Request pg. 48)*

NextStep provides the teaching that forms the basis of a SNQ for the **'854 patent** . Further, NextStep was not before the Examiner during the prosecution of the **'854 patent** and there is a substantial likelihood that a reasonable examiner would consider this teaching important in deciding whether **claims 1-24** are patentable.

Accordingly, **NextStep** raises an SNQ as to **claims 1-24**, which question has not been decided in a previous examination of the **'854 patent**.

Conclusion

10. **All** correspondence relating to this reexamination proceeding should be directed:

By Mail to: Mail Stop Ex Parte Reexam
Attn: Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

11. Registered users of EFS-Web may alternatively submit such correspondence via the electronic filing system EFS-Web, at:

https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html.

EFS-Web offers the benefit of quick submission to the particular area of the Office that needs to act on the correspondence. Also, EFS-Web submissions are "soft scanned" (i.e., electronically uploaded) directly into the official file for the reexamination proceeding, which offers parties the opportunity to review the content of their submissions after the "soft scanning" process is complete.

12. Any inquiry concerning this communication or earlier communications from the examiner, or as to the status of this proceeding, should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

Signed:

/Deandra M. Hughes/
Primary Examiner, CRU 3992

Conferees: ESK

CL

Substitute for form 1449/PTO
(Revised 07/2007)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Complete if Known	
Application Number	~~08/108,877~~ 90/010,965
Filing Date	October 26, 1993
Patent Number	5,455,854 (Exhibit A)
Issue Date	October 3, 1995
First Named Inventor	Dilts
Art Unit	~~2601~~ AU 3992
Examiner Name	~~H. Hong~~ Deandra M. Hughes
Attorney Docket Number	0919/01030

Sheet | 1 | of | 1

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document Number Number - Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages of Relevant Figures Appear
/DMH/	1	US-4,625,081 (Exhibit G)	11-25-1986	Lotito et al.	

OTHER DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s) , volume-issue number(s), publisher, city and/or country where published.	English Language Translation Attached
/DMH/	2	SIXTENSSON et al., Reuse in the Telecommunication Domain Using Object Oriented Technology and ADA, June 1990, Pages 231-239, Washington Ada Symposium Proceedings (Exhibit B)	
/DMH/	3	COHEN et al., Version Management in Gypsy, 1988, Pages 201-215, ACM (Exhibit C)	
/DMH/	4	NEXTSTEP, NeXTStep™ Object-Oriented Programming and The Objective C Language, April 1993, Release 3, pages xi-xvi, 1, 5 and 8-18, Addison-Wesley Publishing Company, Inc. (Exhibit D)	
/DMH/	5	NEXTSTEP, NeXTStep™ Programming Interface Summary, April 1993, Release 3, pages Intro-1, 13 (1-7), Addison-Wesley Publishing Company, Inc. (Exhibit E)	
/DMH/	6	NEXTSTEP, NeXTStep™ General Reference, November 1992, Volume 2, Addison-Wesley Publishing Company, Inc., pages 13-(1-14, 16-36 and 38). (Exhibit F)	
/DMH/	7	ADOBE SYSTEMS INCORPORATED, Programming the Display PostScript® System with NeXTStep™, November 1991, Addison-Wesley Publishing Company, Inc., pages xxii-xxiii, 1-4, 7 and 10. (Exhibit H)	

Examiner Signature	/Deandra M. Hughes/	Date Considered	July 11, 2010

*Examiner: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

EXHIBIT 8

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Ording
U.S. Patent No.:	7,469,381
Issued:	December 23, 2008
Group Art Unit:	2174
Serial No:	11/956,969
Examiner:	B. Pesin
Filed:	December 14, 2007
For:	LIST SCROLLING AND DOCUMENT TRANSLATION, SCALING, AND ROTATION ON A TOUCH-SCREEN DISPLAY
Attorney Docket No.	0919/01028

April 28, 2010

Mail Stop *Ex Parte* Reexam
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

REQUEST FOR REEXAMINATION

Reexamination of United States Patent 7,469,381 (hereinafter, "the '381 patent"), which issued December 23, 2008 to Ording is requested under 35 U.S.C. §§ 302-307, and under 37 C.F.R. § 1.510. This patent is still in force.[1] A copy of the patent in accordance with 37 C.F.R. § 1.510(b)(4) is submitted herewith as Exhibit A. Related continuation applications are pending[2].

I. Claims for which Reexamination is Requested

The '381 patent describes a computer-implemented method according to which an electronic document displayed on a touch screen may be translated to display different portions of the document, and if an edge of the document is reached while translating, an area beyond the edge of

[1] Indeed, a counterclaim for alleged infringement of the '381 patent has been filed in the U.S. District Court for the District of Delaware, *Nokia Corp. v. Apple Inc.*, Case No. 1:09-cv-00791-GMS. That litigation is in its early stages and no discovery regarding the '381 patent has taken place. If the litigation proceeds, third party requester expects there will be a challenge to the validity of the '381 patent therein.

[2] Application Serial Nos. 12/270,810 filed on 11-13-2008, 12/270,812 filed on 11-13-2008, 12/270,815 filed on 11-13-2008, 12/270,805 filed on 11-13-2008, and 12/270,807 filed on 11-13-2008.

the document is displayed and then no longer displayed, in the particular manner claimed.

Reexamination is requested of all Claims 1-20 of the '381 patent.

II. Statement of Substantial New Questions of Patentability

A. The Subject Matter of Claim 1-20

Claims 1-20 recite:

1. A computer-implemented method, comprising:

a device with a touch screen display: displaying a first portion of an electronic document;

detecting a movement of an object on or near the touch screen display;

in response to detecting the movement, translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion;

in response to an edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display: displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion; and in response to detecting that the object is no longer on or near the touch screen display, translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion.

2. The computer-implemented method of claim 1, wherein the first portion of the electronic document, the second portion of the electronic document, the third portion of the electronic document, and the fourth portion of the electronic document are displayed at the same magnification.

3. The computer-implemented method of claim 1, wherein the movement of the object is on the touch screen display.

4. The computer-implemented method of claim 1, wherein the object is a finger.

5. The computer-implemented method of claim 1, wherein the first direction is a vertical direction, a horizontal direction, or a diagonal direction.

6. The computer-implemented method of claim 1, wherein the electronic document is a web page.

7. The computer-implemented method of claim 1, wherein the electronic document is a digital image.

8. The computer-implemented method of claim 1, wherein the electronic document is a word processing, spreadsheet, email or presentation document.

9. The computer-implemented method of claim 1, wherein the electronic document includes a list of items.

10. The computer-implemented method of claim 1, wherein the second direction is opposite the first direction.

11. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching an edge of the document has an associated speed of translation that corresponds to a speed of movement of the object.

12. The computer-implemented method of claim 1, wherein translating in the first direction is in accordance with a simulation of an equation of motion having friction.

13. The computer-implemented method of claim 1, wherein the area beyond the edge of the document is black, gray, a solid color, or white.

14. The computer-implemented method of claim 1, wherein the area beyond the edge of the document is visually distinct from the document.

15. The computer-implemented method of claim 1, wherein translating the document in the second direction is a damped motion.

16. The computer-implemented method of claim 1, wherein changing from translating in the first direction to translating in the second direction until the area beyond the edge of the document is no longer displayed makes the edge of the electronic document appear to be elastically attached to an edge of the touch screen display or to an edge displayed on the touch screen display.

17. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching the edge of the electronic document has a first associated translating distance that corresponds to a distance of movement of the object prior to reaching the edge of the electronic document; and wherein displaying an area beyond the edge of the electronic document comprises translating the electronic document in the first direction for a second associated translating distance, wherein the second associated translating distance is less than a distance of movement of the object after reaching the edge of the electronic document.

18. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching the edge of the electronic document has a first associated translating speed that

corresponds to a speed of movement of the object, and wherein displaying an area beyond the edge of the electronic document comprises translating the electronic document in the first direction at a second associated translating speed, wherein the second associated translating speed is slower than the first associated translating speed.

19. A device, comprising:

a touch screen display;

one or more processors;

memory; and

one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors, the programs including:

instructions for displaying a first portion of an electronic document;

instructions for detecting a movement of an object on or near the touch screen display;

instructions for translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement;

instructions for displaying an area beyond an edge of the electronic document and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion, in response to the edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and

instructions for translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.

20. A computer readable storage medium having stored therein instructions, which when executed by a device with a touch screen display, cause the device to:

display a first portion of an electronic document;

detect a movement of an object on or near the touch screen display;

translate the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement;

display an area beyond an edge of the electronic document and display a third portion of the electronic document, wherein the third portion is smaller than the first portion, if the edge of the electronic document is reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and

translate the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.

In reexamination, as with any proceeding before the U.S. Patent and Trademark Office ("USPTO"), the terms and phrases of a claim are given their broadest reasonable construction. *E.g., In re American Academy of Science Tech Center*, 367 F.3d 1359, 70 USPQ2d 1827, 1830 (Fed. Cir. 2004) ("During examination, 'claims ... are to be given their broadest reasonable interpretation . . .." (*quoting In re Bond*, 910 F.2d 831, 833, 15 USPQ2d 1566 (Fed. Cir. 1990))).

B. Newly cited Prior Art

The '381 patent matured from a U.S. patent application filed December 14, 2007, and claims priority to the filing dates of U.S. Provisional Patent Application Nos. 60/937,993, "Portable Multifunction Device," filed Jun. 29, 2007; 60/946,971, "List Scrolling and Document Translation, Scaling, and Rotation on a Touch-Screen Display," filed Jun. 28, 2007; 60/945,858, "List Scrolling and Document Translation on a Touch-Screen Display," filed Jun. 22, 2007; 60/879,469, "Portable Multifunction Device," filed Jan. 8, 2007; 60/883,801, "List Scrolling and Document Translation on a Touch-Screen Display," filed Jan. 7, 2007; and 60/879,253, "Portable Multifunction Device," filed Jan. 7, 2007. Therefore, the "Critical Date" for prior art relevant to the claims of the '381 patent, under 35 U.S.C. § 102(b) is no earlier than January 7, 2006, if one of the provisional applications filed on that date fully supports the claims. Third party requester does not reach this question as the prior art asserted herein was published prior to January 7, 2006.

The requester respectfully submits that the prior art, under §§ 102(b) taught or suggested the subject matter of the claims of the '381 patent. More particularly, the requester submits that:

- C. Forlines, C. Shen, B. Buxton, "Glimpse: A Novel Input Model for Multi-Level Devices, CHI '05 (Conference on Human Factors in Computing Systems) extended abstracts on Human factors in computing systems (Association for Computing Machinery 2005) pages 1375-78 ("the Glimpse article") (Exhibit B);

in view of :

- M. Millhollon, K. Murray, Microsoft Office Word 2003 Inside Out (Microsoft Press 2004) pages 13-16, 93, 762-65, 802-04 ("Inside Out") (Exhibit C);

and for some proposed grounds of rejection:

- U.S. Patent Application Publication No. 2005/0195154 to Robbins et al. ("the Robbins application") (Exhibit D);
- U.S. Patent No. 6,690,387 to Zimmerman et al. ("the Zimmerman patent") (Exhibit E).

rendered the subject matter of the claims of the '381 patent obvious to one of ordinary skill in the relevant art.

Furthermore, the requester notes that the Glimpse Article, Inside Out, and the Robbins application were not listed on the face of the '381 patent. Consequently, the Glimpse Article, Inside Out, and the Robbins application are newly applied and unquestionably raise new questions of patentability.

C. Basis for Substantial New Questions of Patentability

Claims 1-20 of the '381 patent do not patentably distinguish over combinations of the above-noted newly cited references. In summary, the Glimpse article discloses a computer-implemented navigation method for a Tablet PC touch screen in which a user can, by finger or stylus, without

disengagement from the screen, (a) view an initial (first) portion of an electronic document, (b) translate the document from the first portion to display a different second portion, which can be selected to be any portion of the document, (c) store the second portion in an undo stack, (d) translate the document to display a third portion, and (e) release contact with the screen, whereupon the system automatically restores the view stored in the undo stack, that is, the second portion, which is different from the initial view. (Exhibit B, pp. 1375-78).

Inside Out discloses features of Microsoft Word 2003, including the well-known Print Layout View of electronic documents, which it recommends for Tablet PCs. In Print Layout View, upon scrolling to an edge of the document, an area beyond the edge is displayed in a manner visually contrasting with the document. (Exhibit C, pp. 762-65).

The Robbins application discloses moving from one view to another on a touch screen device by "spring-loaded" animation, and animating back to the initial view upon release, which implies retracing the original motion, as in stretching a spring and then allowing it to retract in the opposite direction. (Exhibit D, ¶¶ 9, 71, 75, 86).

The Zimmerman patent discloses that the panning speed of an electronic document (a list is a disclosed example) corresponds to the speed of motion of a user's finger on the touch screen while the user maintains contact with the screen. When the user breaks contact, the list continues translating but the speed then decreases at a controlled rate until it reaches zero or a predetermined minimum speed (Exhibit E, Abstract; col. 4 ln. 7-37), a motion those skilled in the art would have recognized as a damped motion. (The '381 patent specification refers at column 20, lines 37-46 to: "simulation of a physical device having friction, i.e., damped motion . . . .")

Thus, the Glimpse article discloses or would have rendered obvious each element of Claims 1- 20 of the '381 patent, implemented on a touch screen device, except displaying an area beyond an edge of the document in response to reaching the edge, damping translation speed in certain

circumstances, and certain additional dependent claim limitations discussed below. Inside Out discloses displaying an area beyond an edge of the document in response to reaching the edge while translating the document, on a touch screen device. One skilled in the art would have found a clear motivation to modify the Glimpse article with the teachings of Inside Out, since both references refer to using their teachings on a TabletPC, a touch screen device. The result of applying the described teaching from Inside Out to the Glimpse article's method was predictable, namely, on translating and reaching an edge of a document (for example by a user's finger), an area beyond the edge would be displayed until the user or the system translated the document to display a portion away from the edge. The Robbins application's spring-loaded return animation provides further motivation to reverse direction when returning to a previously viewed portion of the document on disengaging the finger from the screen.

Modifying the Glimpse article method in view of Inside Out would have been obvious because it would have been merely the application of a known technique (displaying an area beyond the edge when the edge is reached during translation) to a known method (the Glimpse method that meets all the other claim limitations but fails to specify what should happen on reaching an edge) to achieve a predictable result. Further, one skilled in the art had only a finite number of choices of known techniques for displaying a document upon reaching an edge of the document. One could simply stop the document, display some sort of separate flag or visual to indicate the edge had been reached, or display an area beyond the edge, as taught by Inside Out and well known in the use of Microsoft Word's print layout view. It would have been obvious to try the Inside Out approach when reaching the edge of a document in a document display and navigation system according to the Glimpse article.[3]

[3] MPEP § 2143 (D), (E) Examples of Basic Requirements of a *Prima Facie* Case of Obviousness, citing *KSR International Co. v. Teleflex Inc.*, 550 U.S. 398, 82 USPQ2d 1385, 1395-97 (2007).

Despite the failure of the original prosecution history to explicitly disclose the examiner's reasons for rejecting the original claims, the examiner clearly considered prior art[4] that rendered unpatentable a claim requiring displaying an area beyond the edge upon translating a document to reach the edge, and, when the object is no longer detected on or near the screen, translating the document in a second direction until the area beyond the edge is no longer displayed. It is clear also from the same prior art of record that the examiner did not believe that the so-called appearance of elastic attachment of the document edge to an edge of the screen, or an edge displayed on the screen, provided patentability. The combination of the Glimpse article and Inside Out provides what apparently the examiner believed was missing in the prior art, namely, displaying: a second portion different from the first portion, a third portion smaller than the first portion, and a fourth portion different from the first portion, in combination with the other claim elements.

It would have been obvious to modify the foregoing combination of the Glimpse article and Inside Out further in view of the Zimmerman patent by incorporating Zimmerman's teaching to track finger translation speed of a digital document on a touch screen display at a speed corresponding to the speed of a finger, and then to damp the speed of translation when the finger breaks contact with the screen.

Because the Glimpse Article, Inside Out, and the Robbins application were not previously considered and are not cumulative of any reference previously considered, combination of these references necessarily raises a new, and not cumulative, question of patentability. Consequently, Reexamination of Claims 1-20 of the '381 patent must be ordered and the claims rejected.

[4] See listing of art discussed in interviews on Examiner-Initiated Interview Summaries for interviews held 6/2/2008; 6/30/2008; and 8/4/2008 in Serial No. 11/956,969.

D. Application of Prior Art References to Claims 1-20

1. Content of the Prior Art

(a) The Glimpse Article

The Glimpse article discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen:

> We describe a technique that supports the previewing of navigation, exploration, and editing operations by providing convenient Undo for unsuccessful and/or undesirable actions on multi-level input devices such as touch screens and pen-based computers. (Abstract, p. i)
>
> Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to pan to other portions of the document, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)
>
> We have used both a TabletPC and a touch sensitive DiamondTouch surface as our pressure sensitive input device. (p. 1377)
>
> As shown in Figure 3, our method replaces Figure 1's State 1 with a new state, which we call Glimpse. When an object is selected for editing through light pressure input, the system enters the Glimpse state and the current value of the property being edited is saved to memory [hereinafter Glimpse buffer] separate from the system's undo stack. This light pressure input indicates intent to edit the selected object. While the user continues to manipulate the object using light pressure input, the system responds by previewing the results of their action. (p. 1376-77)
>
> When editing is finished, the user can either reject or accept the edit by performing one of two actions. If the user lifts their finger or stylus (or otherwise releases the input), the system returns to State 0 and the edit is automatically 'undone' by retrieving the saved state. When possible ***we animate this undo graphically*** so that the action is as clear to the user as possible. If the user increases the pressure of their input past a certain threshold, the system enters State 2 and the previewed changes to the edited object becomes the object's current state. In this transition, the previously saved values of the object are pushed onto the system's undo stack. While the user remains in State 2, changes to the object are saved as

they occur. Reentering the Glimpse state from State 2 again stores the current value of the object being edited to memory. The Glimpse state previews the further change of this value, which can again be confirmed by reentering State 2. (p. 1377) (emphasis added).

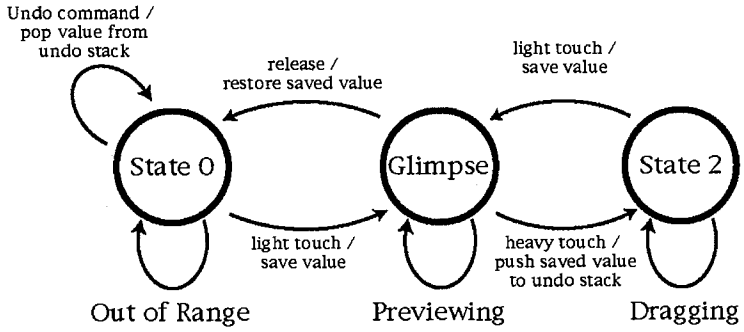


Figure 3. Glimpse enabled transition diagram for pressure sensitive direct input devices.

Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch:

> Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to pan to other portions of the document, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)

The Glimpse article teaches panning and zooming techniques for navigating an electronic document. Glimpse teaches using these features separately, allowing a user to pan to different areas of a document while maintaining the same magnification. The following passage and the foregoing passage illustrate this functionality:

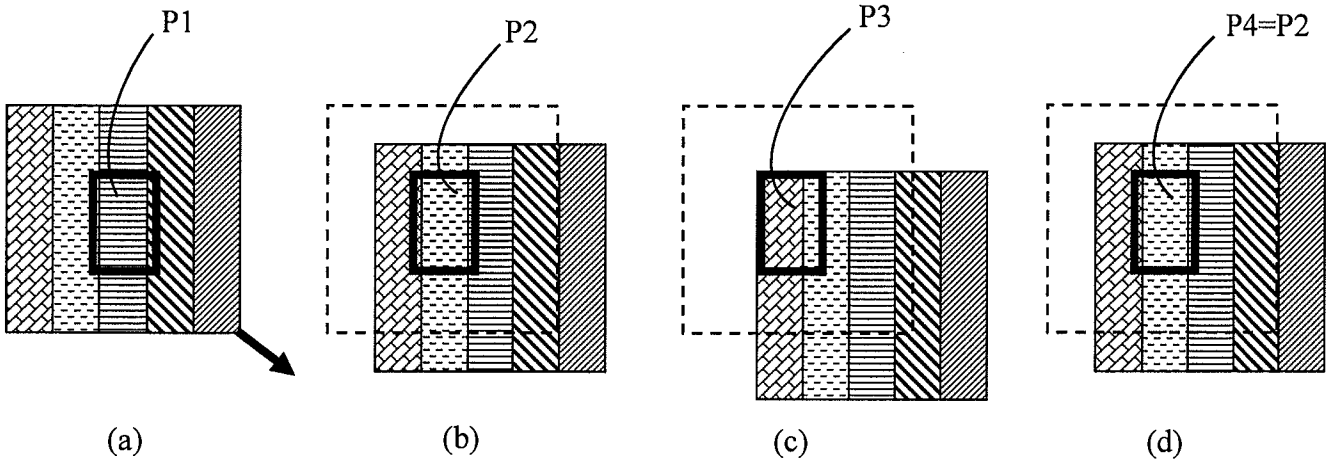
> When navigating through a dataset using a pan and zoom interface, one often wants to temporarily zoom-in in order to take a more detailed look at some portion of the data before returning to the current zoom level. Using a traditional interface, zoom-in and zoom-out are separate commands (and may require the user to traverse to a tool pallet in order to switch tools). Furthermore, if zooming does not occur in fixed increments, inaccuracies in the operation of the zoom tool can make the task of returning to an exact zoom level difficult if not impossible. Similarly, for drag-to-pan movement around a dataset, retracing one's path in order to return to a previous location can be very difficult. It is a combination of these two

> difficulties that cause many users to complain that they become "lost" in the dataset when using a pan-and-zoom interface. (p. 1377).

Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen:

> Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added).

A mode of operation of the Glimpse system and method is shown in the sequence of diagrams below (provided by the third party requester). In the diagrams, the small black rectangle represents the screen of a touch screen device, held stationary by the user. The diagrams show the striped document in its entirety, although only a portion is visible on the screen. When the user drags her finger across the screen, the document "sticks" to the finger and moves in the same direction relative to the screen. The dashed box marks the initial position of the document.



Thus, when viewing an initial Portion P1 of an object such as a document as shown in diagram (a), the user could translate the document by moving a finger with a light touch on the screen (within the small black box) until a user-selected Portion P2, different from P1, would have been visible on the screen as shown in diagram (b). While viewing P2, and maintaining contact with the screen, the user could have pressed harder to move P2 from the Glimpse buffer memory into the system undo stack. Without breaking contact, the user could have lightened her touch to return to the glimpse preview mode, and continued translating the document until a third user-

selected Portion P3 would have been viewed as shown in diagram (c). Upon then sensing a breaking of contact with the screen, the system would have restored the view to that stored in the system undo stack, which in this example is P2, as shown in diagram (d). The preferred way to accomplish this "undo" in Glimpse system is to animate the return graphically.

(b) Inside Out

The Inside Out reference teaches that Microsoft Word 2003 can display a document in several different views and that print layout view is the "recommended" view for Tablet PCs running Microsoft Word 2003. (p. 764).

> If you're using a Tablet PC, you can add ink annotations directly on top of content in documents. For instance, you can circle text, draw arrows on graphics, highlight key topics, or cross out chart elements. You must be running Word 2003 on a Tablet PC to use ink annotations, and it is recommended that you work in Print Layout view for optimal results. (p. 764).

In Microsoft Word 2003's print layout view, the edge of a document and a background area beyond the edge of the document are visible. The following screenshot published in Inside Out illustrates this type of view and again highlights the integration with a Tablet PC:

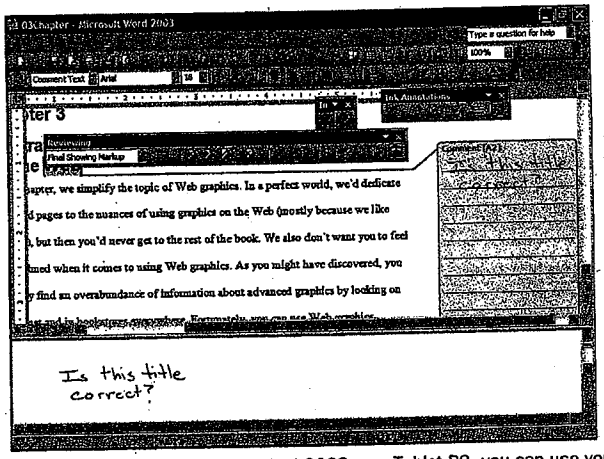


Figure 27-10. If you're using Word 2003 on a Tablet PC, you can use your tablet pen to add ink comments. After you add ink comments, others can view your comments on other types of systems.

Inside Out, p. 762.

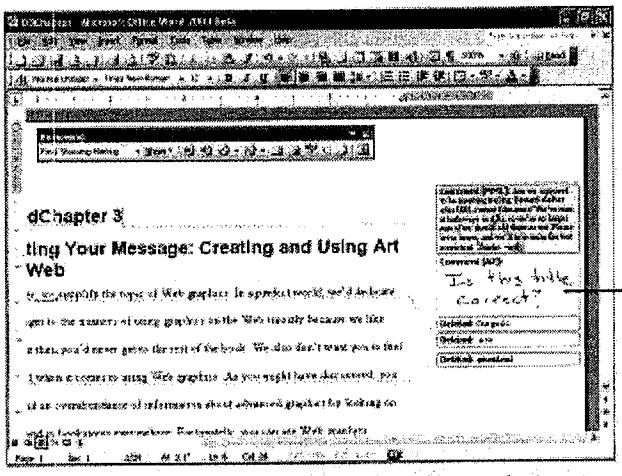


Figure 27-13. Ink comments appear alongside standard comments and tracked changes in a document's margin or Reviewing Pane. You can copy and delete ink comments, but you can't add text.

Inside Out, p. 764.

These screen shots show that in print layout view, when a background beyond the edge of a document is displayed after panning to the edge, there is less of the document on the screen (third portion) than there was before the user panned to the edge (first portion).

Inside Out also teaches that a user can zoom in on a region of the document such that the edge of the document and the area beyond the edge are not displayed on the screen initially:

> Zooming in on information: You can increase the viewing size of your document by using the Magnifier button (which displays the document at actual size) or by indicating a size in the Zoom box (either by selecting a size in the Zoom list or by typing a percentage value). To zoom in on a selected area, click the Magnifier button, and then click in the area of the document you want to examine more closely. You can use the Zoom box to further modify your view, if necessary. (p. 93).

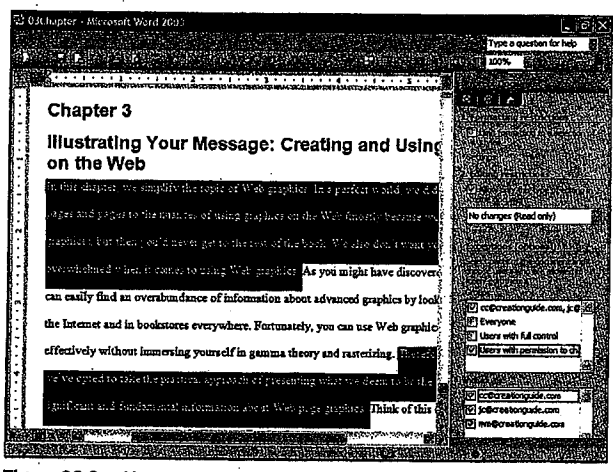


Figure 28-9. You can control the amount of editing allowed in a document by selecting areas in the document that can be edited and then assigning which users can edit which content areas.

Inside Out, p. 803.

As can be seen in Fig. 28-9, the document extends beyond the edge of the screen in every direction. Thus, the entire display is filled with the document (first portion). The user then pans in any direction to reveal other regions of the document. If the user reaches the edge of the document, the background area beyond the edge becomes visible and a smaller portion of the document (third portion) is displayed on the screen:

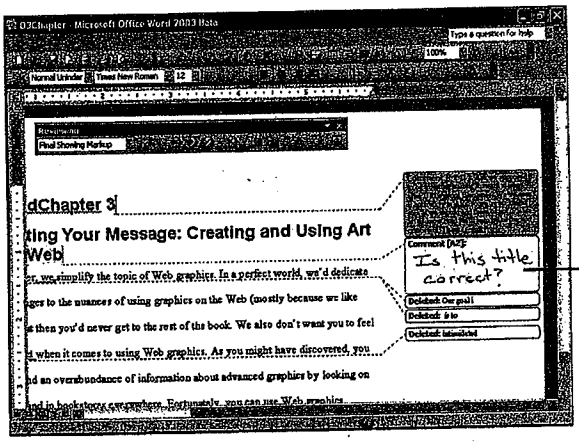


Figure 27-13. Ink comments appear alongside standard comments and tracked changes in a document's margin or Reviewing Pane. You can copy and delete ink comments, but you can't add text.

Inside Out, p. 764.

(c) The Robbins Application

The Robbins application No. 2005/0195154, published September 8, 2005, teaches panning and zooming on a portable device:

> The present invention relates to a system and/or methodology that facilitate navigating and/or browsing large information spaces on relatively small portable devices such as portable phones, PDAs and the like, for example. In particular, the system and method allow navigation of multi-resolution graphical content at multiple levels of magnification. (¶ 4).

Robbins also teaches using a touch screen as the input for the portable device:

> [T]he portable device can have a touch screen or some other type of display screen or touch pad that is sensitive to and/or receptive to a pointing device. (¶ 7; see also ¶100).

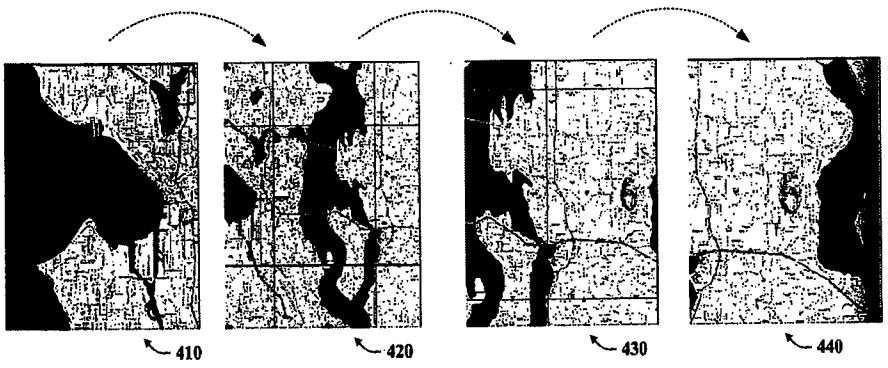
Robbins teaches "glancing" at other regions of the document and using animations to transition between regions:

> According to still another aspect of the invention, a user can "glance" at other parts of a data-set or document while browsing through such data-set on a portable device. For example, imagine that a user indicates an area of the data-set for detailed inspection. In general, this may happen via clicking on a region, selecting the name of a region from a menu or dialog, or pressing a hardware button or function key that is assigned ahead of time to a particular region. When the user wants to quickly glance at another region, the user can instruct the application to temporarily switch the view to another region by again selecting another region via the aforementioned techniques. After a time-delay or after the user releases a hardware or software button, the view quickly and smoothly (e.g., via animation) can snap back to the previous view. (¶ 9).

Robbins explains one possible animation for this "snap back," teaching that when panning from one region to another, the screen will display the first region then zoom out while simultaneously panning toward the new region to be displayed. As the screen approaches the new region, the view zooms in to align the new area with the edges of the screen. This functionality is explained in the passage and figure below:

> For example, in FIG. 4, a series of screen views of the map illustrate a ***smooth and/or animated transition by panning from sector 4 to sector 6***, the sibling view of sector 4. In particular, screen view 410 shows a zoomed in view of sector or sub-sector 4. However, when panning from sub-sector 4 to sub-sector 6, the screen view zooms out (420) and then gradually zooms in (430) as sub-sector 6 is reached. When sub-sector 6 is in "full" view to the near exclusion of other sub-sectors (enlarged sub-sector 6 takes up the display space), the sub-sector 6 appears as enlarged or zoomed in according to screen view 440 (e.g., enlarged to a similar degree as the initial focus of interest: sub-sector 4). All of these view

transitions (e.g., in, out, and same-level translation) are animated smoothly by using a simplified version of a pan-and-zoom algorithm. (¶ 75) (emphasis added).



VIEW ZOOMS OUT DURING PAN FROM ONE SIBLING VIEW (SECTOR 4) TO ANOTHER (SECTOR 6)

FIG. 4

Robbins teaches using this panning animation after a user stops glancing at a region to "snap back to the previous view." (¶ 9). This function can be provided by Robbins' system while a user is moving dynamically around a grid of information. (¶ 92).

Robbins further describes its spring-loaded glance and return function in ¶86:

> To "glance" momentarily in another direction (at a nearby view) the user presses-and-holds down on the appropriate number key. When the key is released, the view animates back to the previous view. This spring-loaded glancing can be extended to also work with child views of the current view. If the user is currently zoomed out, such that segment cues are shown for the current view's child segments, pressing and holding on the number key will temporarily zoom the view to the appropriate child view. Releasing that same key will then return to the parent view. This spring-loaded view shifting allows the user to quickly glance at other sections of the data-set without losing track of their preferred center of interest.

The animation provided in Robbins can include following a route. (¶107).

Robbins also teaches that the amount of detail displayed during the panning transition depends on the speed of the pointing device that initiated the panning:

> Turning now to FIGS. 16-24, a navigational sequence using a pointing device on a small portable device is shown, wherein each figure represents a phase in the sequence. Looking initially at FIG. 16, there is illustrated an image of a portable device 1600 displaying a

portion of a map 1610 on its screen. In general, as the speed of a pointing device increases, less detail (e.g., more of an overview) of the underlying content appears on the screen. However, at slower speeds, more detail of the underlying content is displayed. Transitions between views of the content are smooth and fluid-like rather than abrupt zoom-in and out changes. (¶ 112).

(d) The Zimmerman Patent

The Zimmerman patent discloses that the panning speed of an electronic document (a list is a disclosed example) corresponds to the speed of motion of a user's finger on the touch screen while the user maintains contact with the screen, noting that in a natural manner, the initial speed of displacement of the displayed image corresponds to the speed of motion of the finger along the screen. When the user breaks contact, the list continues translating but the speed is then slowly decreased until it reaches zero, which one skilled in the art would understand to be a damped motion. (Exhibit E, Abstract; and col. 3, ln. 45 - col. 4, ln. 37). (The '854 patent specification refers at column 20, lines 37-46 to: "simulation of a physical device having friction, i.e., damped motion" . . .).

2. Grounds for Rejection of the Claims

(a) ***First Ground for Rejection***: Claims 1-11, 13, 14, 16, 17, 19, and 20 would have been obvious over the Glimpse article in view of Inside Out.

By comparing the content of the Glimpse article and Inside Out to Claims 1-11, 13, 14, 16, 17, 19, and 20 of the '381 patent, it will become clear that to one of ordinary skill in the art the claimed subject matter would have been obvious.

Claim	**Prior Art**
1[a] A computer implemented method, comprising:	The Glimpse article discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract).
1[b] at a device with a touch screen display:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762).

1[c] displaying a first portion of an electronic document	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
1[d] detecting a movement of an object on or near the touch screen display	Glimpse teaches detecting movement of an object on the touch screen display by using a pressure sensitive touch screen to identify multi-level input, including both light touch and heavy touch: The technique we propose provides a method for editing objects with a multi-level input device such as a pressure sensitive stylus, ***pressure sensitive touch screen***, or pop-through mouse. We have used both a TabletPC and a touch sensitive DiamondTouch [3] surface as our pressure sensitive input device. (p. 1376)(emphasis added). Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen: Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added). Inside Out discloses stylus input to a touch screen Tablet PC. (Fig. 27-10, p. 762).
1[e] in response to detecting the movement, translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77). In addition, the 381 patent admits that the prior art teaches this

	limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
1[f] in response to an edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display: displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion; and	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.
1[g] in response to detecting that the object is no longer on or near the touch screen display, translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Clearly, continuing in the first direction is not an operable option for accomplishing the return. Thus, Glimpse translates in another direction until the area is no longer displayed, and then may or may not continue translating, depending on the location of the saved state of the document. In either case, the claim language reads on the Glimpse process.

	If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion occupies the viewing area of the screen and obscures the previously displayed area beyond the edge.
2. The computer-implemented method of claim 1, wherein the first portion of the electronic document, the second portion of the electronic document, the third portion of the electronic document, and the fourth portion of the electronic document are displayed at the same magnification.	Glimpse teaches panning and zooming techniques for navigating an electronic document. Glimpse teaches using these features separately, allowing a user to pan to different areas of a document while maintaining the same magnification. (p. 1377)
3. The computer-implemented method of claim 1, wherein the movement of the object is on the touch screen display.	Glimpse teaches tracking touch input on a pressure sensitive screen to provide previewing by changing the portion of the document viewed on the touch screen. (p. 1376-77)
4. The computer-implemented method of claim 1, wherein the object is a finger.	Glimpse discloses a pressure sensitive screen that can sense a user's finger: Any multi-state input device that also provides tracking (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus ***or finger***) can exploit this technique. (p. 1376) (emphasis added).
5. The computer-implemented method of claim 1, wherein the first direction is a vertical direction, a horizontal direction, or a diagonal direction.	Glimpse discloses panning to different portions of an electronic document without any restriction as to the direction of the panning: We describe a technique that supports the previewing of navigation, exploration, and editing operations by providing convenient Undo for unsuccessful and/or undesirable actions on multi-level input devices such as touch screens and pen-based computers. (Abstract). [A] user may click and drag using light input to pan to other portions of the document, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377). To the extent vertical, horizontal, or diagonal tracking are not explicitly disclosed in Glimpse, they would have been obvious to

	one skilled in the art as a matter of common sense and as a finite set of options to try, each having a predictable result.
6. The computer-implemented method of claim 1, wherein the electronic document is a web page.	It would have been obvious to a skilled person to provide the functionality of Glimpse in view of Inside Out in the case of viewing a web page, as a matter of general knowledge and common practice.
7. The computer-implemented method of claim 1, wherein the electronic document is a digital image.	The electronic documents described and/or shown in the Glimpse article and Inside Out are digital images, as would have been appreciated by a skilled person.
8. The computer-implemented method of claim 1, wherein the electronic document is a word processing, spreadsheet, email or presentation document.	The Glimpse article provides a "system wide" method for systems such as Microsoft Windows OS and describes how the authors method would have been applied in the preparation of the article, obviously using a word processing program generating a word processing digital document. (p. 1375-78) Inside Out shows and describes use of the print layout view on a Tablet PC for editing a Microsoft Word digital document. (p. 764)
9. The computer-implemented method of claim 1, wherein the electronic document includes a list of items.	Glimpse discloses that it was well known in the art to navigate lists of items: Ramos, et al. [4] described a continuous pressure-sensing stylus to manipulate multi-state objects. They mapped continuous pressure to visual properties of the pointer, e.g., moving the pointer down a ***list of menu selections*** as pressure increases, or to change the appearance of objects, e.g., making objects larger and smaller based on pressure. (p. 1376) (emphasis added).
10. The computer-implemented method of claim 1, wherein the second direction is opposite the first direction.	As described above, Glimpse animates the undo operation graphically in returning to a previously viewed portion. (p. 1377). To do so involves a selection from a finite set of possible animation techniques. It would have been obvious to select return panning in the opposite direction from this finite set.
11. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching an edge of the document has an associated speed of translation that corresponds to a speed of movement of the object	The Glimpse article refers to "any multi-state input device that also provides tracking . . . as in the case of a stylus of finger . . ." (p. 1376). A skilled person would read this to mean translating a document at the same speed as a finger or stylus. It would have been obvious to use tracking in the Glimpse system as modified according to Inside Out.
13. The computer-implemented method of claim 1, wherein the area beyond the edge of the document is black, gray, a solid color, or white.	Inside Out, Fig. 27-13, shows a gray area beyond the edge of a displayed digital document on a touch screen. It would have been obvious to utilize this in modifying the Glimpse article according to Inside Out. To select another color also would have been obvious.
14. The computer-implemented method of claim 1, wherein the area beyond the	Inside Out, Fig. 27-13, shows a visually distinct area beyond the edge of a displayed digital document on a touch screen. It would

edge of the document is visually distinct from the document.	have been obvious to utilize this in modifying the Glimpse article according to Inside Out.
16. The computer-implemented method of claim 1, wherein changing from translating in the first direction to translating in the second direction until the area beyond the edge of the document is no longer displayed makes the edge of the electronic document appear to be elastically attached to an edge of the touch screen display or to an edge displayed on the touch screen display.	As shown graphically below, Glimpse allows a user to save to the system undo stack a view in which the edge of the document is even with the edge of the screen of a touch screen device. Then, when the user further translates in a preview mode to expose an area beyond the edge as would have been obvious in view of Inside Out, and then releases contact with the screen, the view "snaps" back to that stored in the undo stack, making the edge of the document appear to be elastically attached to the edge of the screen display.
17 [a] The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching the edge of the electronic document has a first associated translating distance that corresponds to a distance of movement of the object prior to reaching the edge of the electronic document;	The Glimpse article references tracking of the stylus or finger object on a touch screen (p. 1376-77) and the translation distance prior to reaching an edge clearly can be large in comparison to dimensions of the screen.
17[b] and wherein displaying an area beyond the edge of the electronic document comprises translating the electronic document in the first direction for a second associated translating distance, wherein the second associated translating distance is less than a distance of movement of the object after reaching the edge of the electronic document.	Inside Out shows a small dimension of area beyond the edges, obviously often less than the distance the document has been translated before reaching the edge. (Fig. 27-13, p. 764). It would have been obvious to modify Glimpse by exposing a relatively small area beyond the edges as shown in Inside Out, where the movement required to expose that area is less than the distance the document was translated before reaching the edge.
19[a] A device, comprising:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract).
19[b] a touch screen display;	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762)
19[c] one or more processors; memory; and one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors, the programs including:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Glimpse also discloses using a "Tablet PC…as our pressure sensitive input device." (p.1376) Inside Out discloses a Tablet PC. One skilled in the art would have known that these devices contain a processor, memory, and programs stored in the memory and configured to be executed by the processor.

19[d] instructions for displaying a first portion of an electronic document;	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
19[e] instructions for detecting a movement of an object on or near the touch screen display;	Glimpse teaches detecting movement of an object on the touch screen display by viewing a pressure sensitive touch screen to identify multi-level input, including both light touch and heavy touch: The technique we propose provides a method for editing objects with a multi-level input device such as a pressure sensitive stylus, ***pressure sensitive touch screen***, or pop-through mouse. We have used both a TabletPC and a touch sensitive DiamondTouch [3] surface as our pressure sensitive input device. (p. 1376)(emphasis added). Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen: Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added). Inside Out discloses stylus input to a touch screen Tablet PC. (Fig. 27-10, p. 762)
19[f] instructions for translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement;	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77)

	In addition, the 381 patent admits that the prior art teaches this limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
19[g] instructions for displaying an area beyond an edge of the electronic document and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion, in response to the edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.
19[h] instructions for translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Continuing in the first direction is not an operable option for accomplishing the return. If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the

	location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion prior reaching an edge obscures the previously displayed area beyond the edge.
20[a] A computer readable storage medium having stored therein instructions, which when executed by a device with a touch screen display, cause the device to:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762)
20[b] display a first portion of an electronic document;	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
20[c] detect a movement of an object on or near the touch screen display;	Glimpse teaches detecting movement of an object on the touch screen display by using a pressure sensitive touch screen to identify multi-level input, including both light touch and heavy touch: The technique we propose provides a method for editing objects with a multi-level input device such as a pressure sensitive stylus, ***pressure sensitive touch screen***, or pop-through mouse. We have used both a TabletPC and a touch sensitive DiamondTouch [3] surface as our pressure sensitive input device. (p. 1376)(emphasis added). Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen: Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the popthrough mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added).

	Inside Out discloses stylus input to a touch screen Tablet PC. (Fig. 27-10, p. 762)
20[d] translate the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77) In addition, the 381 patent admits that the prior art teaches this limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: > As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
20[e] display an area beyond an edge of the electronic document and display a third portion of the electronic document, wherein the third portion is smaller than the first portion, if the edge of the electronic document is reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.

20[f] translate the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Continuing in the first direction is not an operable option for accomplishing the return. If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion prior reaching an edge obscures the previously displayed area beyond the edge.

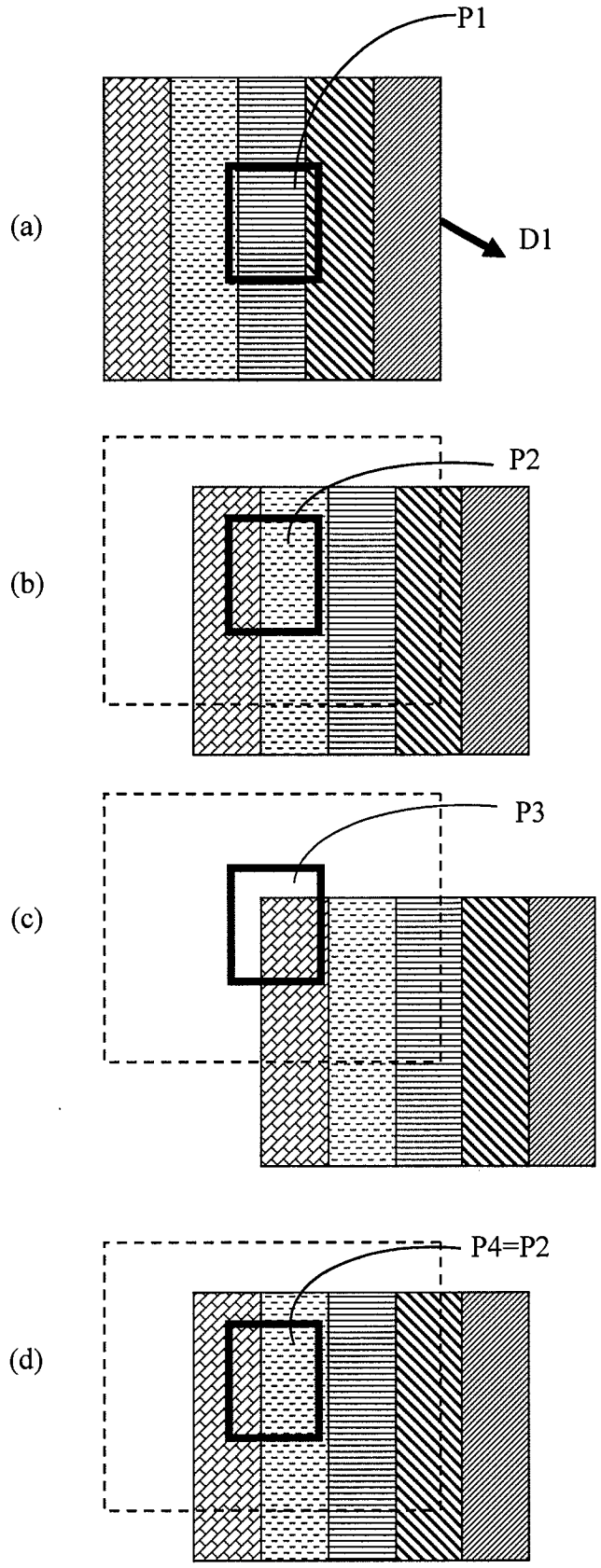
The Glimpse article discloses all of the elements of Claim 1 of the '381 patent implemented on a touch screen device except displaying an area beyond an edge of the document in response to reaching the edge. As Glimpse teaches an animated return to a previously viewed portion of a document different from the initially viewed portion, it would have been obvious for the return animation to be translating back to the previously viewed portion, which necessarily would involve movement in a different direction. Inside Out discloses displaying an area beyond an edge of the document in response to reaching the edge, on a touch screen device. One skilled in the art would have found a clear motivation to modify the Glimpse article with the teachings of Inside Out, as they both disclose using a Tablet PC as a touch screen input device for their navigating and display techniques. The result of applying the described teaching from Inside Out to the Glimpse article method was predictable, namely, on translating and reaching an edge, an area beyond the edge would be displayed until the user or the system translated the document to display a portion away from the edge. Therefore, it would have been obvious to combine these two references to obtain the invention claimed in Claim 1.

The following diagrams and caption further explain how Claim 1 reads on the Glimpse article as modified in accordance with Inside Out.

DIAGRAM:
GLIMPSE+INSIDE OUT

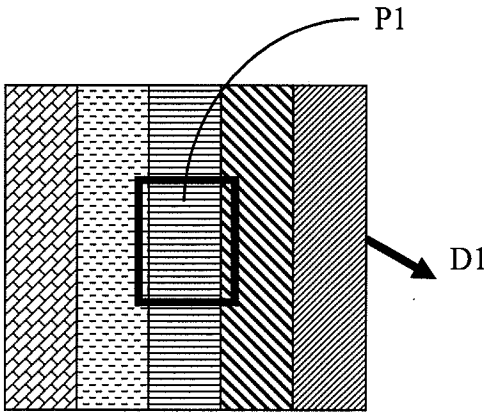
Referring to a combination of the Glimpse article and Inside Out references, the modified Glimpse system and method provide a mode of operation as shown in the sequence of diagrams to the right. In the diagrams, the small black rectangle represents the screen of a touch screen device, held stationary by the user. The diagrams show the striped document in its entirety, although only a portion is visible on the screen. When the user drags her finger across the screen, the document "sticks" to the finger and moves in the same direction. The dashed box marks the initial position of the document.

When initially viewing a Portion P1 of an object such as a document as shown in diagram (a), the user could translate the document by moving a finger with a light touch on the screen until a Portion P2 would have been visible on the screen as shown in diagram (b). P2 is different from P1. While viewing P2, and maintaining contact with the screen, the user could have pressed harder to move P2 from the Glimpse buffer memory into the system undo stack. Without breaking contact, the user could have lightened her touch to return to the glimpse preview mode, and continued translating the document until an edge would have been exposed plus an area beyond the edge in accordance with the teaching of Inside Out, as shown in diagram (c), where the portion of the document being viewed, P3, is smaller than P1. Upon then sensing a breaking of contact with the screen, the system would have restored the view stored in the system undo stack, that is P2. The "fourth portion" of claim 1 reads on this final view P4, the same as P2, which is different from P1. The claim does not require that the fourth portion be different from the second portion.

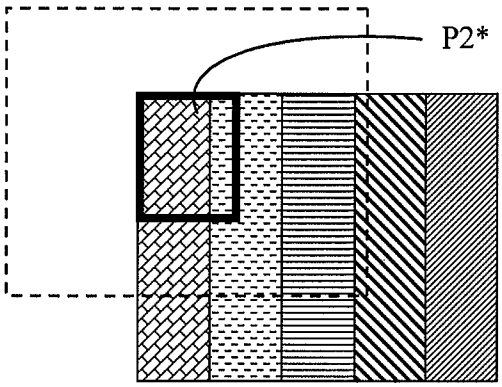


The Glimpse article also supports a mode of operation in which the user can browse in the document to a position in which the portion viewed is aligned with one or more edges of the screen, as shown in diagram (f). Then the user can press harder to store this view P2* in the system undo stack. Then, following further browsing to a position as shown in diagram (g) to view Portion P3, upon the user releasing contact, the system will animate a return to view P2*, a movement just sufficient to no longer expose the area beyond the edge of the document. This corresponds to a fourth portion P4 different from P1.

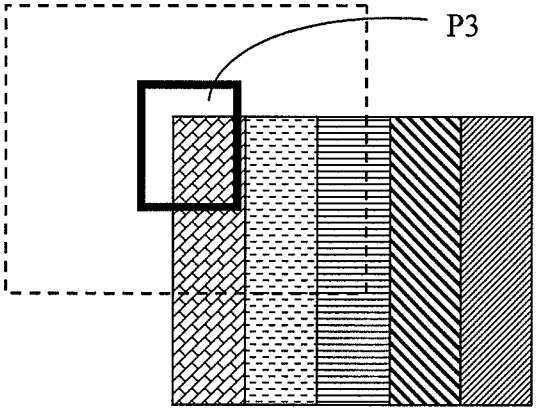
(e)



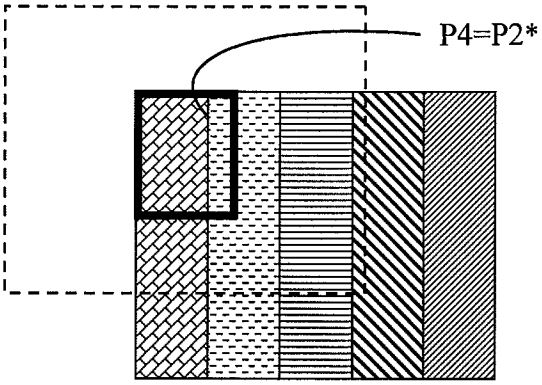
(f)



(g)



(h)



As demonstrated in the chart above, independent claims 19 and 20 are device and computer readable medium versions of claim 1, and both read on the combination of the Glimpse article and Inside Out according the same analysis applied above in connection with claim 1. Dependent claims 2-11, 13, 14, 16, and 17 would have been obvious for the reasons stated in the above claims chart for this first ground of rejection.

(b) ***Second Ground for Rejection***: Claims 1-11, 13-16, 17, 19, and 20 would have been obvious over the Glimpse article in view of Inside Out and the Robbins application.

By comparing the content of the Glimpse article, Inside Out, and the Robbins application to claims 1-11, 13-16, 17, 19, and 20 of the '381 patent, it will become clear that the claimed subject matter would have been obvious to one of ordinary skill in the art.

Claim	Prior Art
1[a] A computer implemented method, comprising:	The Glimpse article discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract).
1[b] at a device with a touch screen display:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762)
1[c] displaying a first portion of an electronic document	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to

Claim	Prior Art
	see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
1[d] detecting a movement of an object on or near the touch screen display	Glimpse teaches detecting movement of an object on the touch screen display by using a pressure sensitive touch screen to identify multi-level input, including both light touch and heavy touch: The technique we propose provides a method for editing objects with a multi-level input device such as a pressure sensitive stylus, ***pressure sensitive touch screen***, or popthrough mouse. We have used both a TabletPC and a touch sensitive DiamondTouch [3] surface as our pressure sensitive input device. (p. 1376)(emphasis added). Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen: Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the popthrough mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added). Inside Out discloses stylus input to a touch screen Tablet PC. (Fig. 27-10, p. 762)
1[e] in response to detecting the movement, translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77) In addition, the 381 patent admits that the prior art teaches this limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen

Claim	Prior Art
	at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
1[f] in response to an edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display: displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion; and	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.
1[g] in response to detecting that the object is no longer on or near the touch screen display, translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Continuing in the first direction is not an operable option for accomplishing the return. If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion prior reaching an edge obscures the previously displayed area beyond the edge. In addition, to the extent (assuming for the sake of argument) panning (translating) in a second direction is not explicitly or

Claim	Prior Art
	inherently disclosed by Glimpse, such action would have been an obvious modification of Glimpse, as modified by Inside Out, in light of the Robbins reference. One of ordinary skill in the art would have been motivated to combine Glimpse with Robbins because both solutions are directed to the problem of navigating electronic documents by panning and zooming. Robbins teaches allowing a user to "glance" at other parts of a document with the option to easily revert back to the previous location. (¶ 9). Glimpse teaches animating a transition from the preview state to the saved state. (p. 1377). Robbins also teaches animating a transition between views while the user is panning. (¶ 75, 86). In fact, Robbins teaches using a spring-loaded panning animation after a user stops glancing at a region to "snap back to the previous view." (¶ 9, 71, 75, 86). Because Glimpse calls for animating a transition, it would have been common sense to choose a simple animation that unwinds the panning and pans back from the preview state to the saved state, as more fully disclosed in Robbins. Thus, it would have been obvious to one of ordinary skill in the art to utilize the animations of the Robbins application in implementing the Glimpse technique for undo animation, as modified by Inside Out.
2. The computer-implemented method of claim 1, wherein the first portion of the electronic document, the second portion of the electronic document, the third portion of the electronic document, and the fourth portion of the electronic document are displayed at the same magnification.	Glimpse teaches panning and zooming techniques for navigating an electronic document. Glimpse teaches using these features separately, allowing a user to pan to different areas of a document while maintaining the same magnification. (p. 1377)
3. The computer-implemented method of claim 1, wherein the movement of the object is on the touch screen display.	Glimpse teaches tracking touch input on a pressure sensitive screen to provide previewing by changing the portion of the document viewed on the touch screen. (p. 1376-77)
4. The computer-implemented method of claim 1, wherein the object is a finger.	Glimpse discloses a pressure sensitive screen that can sense a user's finger: Any multi-state input device that also provides tracking (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus ***or finger***) can exploit this technique. (p. 1376) (emphasis added).
5. The computer-implemented method of claim 1, wherein the first direction is	Glimpse discloses a pressure sensitive screen that can sense a user's finger:

Claim	Prior Art
a vertical direction, a horizontal direction, or a diagonal direction.	Any multi-state input device that also provides tracking (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus ***or finger***) can exploit this technique. (p. 1376) (emphasis added).
6. The computer-implemented method of claim 1, wherein the electronic document is a web page.	The Robbins application teaches navigating a dataset by panning and zooming. The dataset can include a webpage: Referring now to FIG. 2, there is illustrated a block diagram of another advanced navigation system 200 that facilitates the navigation of two-dimensional content space in portable devices. Before navigation (or browsing) can begin, content such as a data-set can be uploaded or accessed by the portable device. The content can include, but is not limited to, any type of document, such as pictures, calendars, images, spreadsheets, reports, maps, books, text, ***web pages***, etc. as well as their related programs or applications. (¶ 67) (emphasis added). It would have been obvious to utilize the combined teachings as described above for web pages, in view of Robbins.
7. The computer-implemented method of claim 1, wherein the electronic document is a digital image.	Robbins teaches navigating a dataset by panning and zooming. The dataset can include an image: Referring now to FIG. 2, there is illustrated a block diagram of another advanced navigation system 200 that facilitates the navigation of two-dimensional content space in portable devices. Before navigation (or browsing) can begin, content such as a data-set can be uploaded or accessed by the portable device. The content can include, but is not limited to, any type of document, such as pictures, calendars, ***images***, spreadsheets, reports, maps, books, text, web pages, etc. as well as their related programs or applications. (¶ 67) (emphasis added). It would have been obvious to utilize the combined teachings as described above for digital images, in view of Robbins.
8. The computer-implemented method of claim 1, wherein the electronic document is a word processing, spreadsheet, email or presentation document.	Robbins also teaches navigating a dataset by panning and zooming. The dataset can include a document or spreadsheet: Referring now to FIG. 2, there is illustrated a block diagram of another advanced navigation system 200 that facilitates the navigation of two-dimensional content space in portable devices. Before navigation (or browsing) can begin, content such as a data-set can be uploaded or accessed by the portable device. The content can include, but is not limited

Claim	Prior Art
	to, any type of document, such as pictures, calendars, images, ***spreadsheets***, reports, maps, books, text, web pages, etc. as well as their related programs or applications. (¶ 67) (emphasis added). It would have been obvious to utilize the combined teachings as described above for spreadsheets, in view of Robbins.
9. The computer-implemented method of claim 1, wherein the electronic document includes a list of items.	Glimpse discloses that it was well known in the art to navigate lists of items: Ramos, et al. [4] described a continuous pressure-sensing stylus to manipulate multi-state objects. They mapped continuous pressure to visual properties of the pointer, e.g., moving the pointer down a ***list of menu selections*** as pressure increases, or to change the appearance of objects, e.g., making objects larger and smaller based on pressure. While this work provides a good exploration of the design space for pressure sensitive widgets, no recommendations are made for implementing pressure sensitivity in a systemwide manner. (p. 1376) (emphasis added).
10. The computer-implemented method of claim 1, wherein the second direction is opposite the first direction.	Robbins also discloses a "spring-loaded" panning from one location to another in an electronic document and then snapping back to the first location using a smooth animation. If the user has directly panned from one location to another, the snapping back movement will simply reverse direction and the second direction will be the opposite of the first. Robbins discloses an example "snap back animation," teaching that when panning from one region to another, the screen will display the first region then zoom out while simultaneously panning toward the new region. As the screen approaches the new region, the view zooms in to align the new area with the edges of the screen. (¶ 75).
11. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching an edge of the document has an associated speed of translation that corresponds to a speed of movement of the object.	Robbins also teaches that the amount of detail displayed during the panning transition depends on the speed of the pointing device that initiated the panning: Turning now to FIGS. 16-24, a navigational sequence using a pointing device on a small portable device is shown, wherein each figure represents a phase in the sequence. Looking initially at FIG. 16, there is illustrated an image of a portable device 1600 displaying a portion of a map 1610 on its screen. In general, as the speed of a pointing device increases, less detail (e.g., more of an overview) of the underlying content appears on the screen. However, at slower speeds, more detail of the underlying content is

Claim	Prior Art
	displayed. Transitions between views of the content are smooth and fluid-like rather than abrupt zoom-in and out changes. (¶ 112). It would have been obvious to modify the combined teachings as described to provide translation speed corresponding to movement of the pointing object, in view of Robbins.
13. The computer-implemented method of claim 1, wherein the area beyond the edge of the document is black, gray, a solid color, or white.	Inside Out, Fig. 27-13, shows a gray area beyond the edge of a displayed digital document on a touch screen. It would have been obvious to utilize this in modifying the Glimpse article according to Inside Out. (p. 764). To select another color also would have been obvious.
14. The computer-implemented method of claim 1, wherein the area beyond the edge of the document is visually distinct from the document.	Inside Out, Fig. 27-13, shows a visually distinct area beyond the edge of a displayed digital document on a touch screen. It would have been obvious to utilize this in modifying the Glimpse article according to Inside Out. (p. 764).
15. The computer-implemented method of claim 1, wherein translating the document in the second direction is a damped motion.	Robbins also discloses animating smoothly when transitioning between viewed portions of an electronic document (¶ 75). It would have been obvious to modify the combined teachings as described to provide smooth (that is, damped) animation when returning to a previous view, in view of Robbins.
16. The computer-implemented method of claim 1, wherein changing from translating in the first direction to translating in the second direction until the area beyond the edge of the document is no longer displayed makes the edge of the electronic document appear to be elastically attached to an edge of the touch screen display or to an edge displayed on the touch screen display.	As shown graphically above in connection with the first ground of rejection, Glimpse allows a user to save to the system undo stack a view in which the edge of the document is even with the edge of the screen of a touch screen device. Then, when the user further translates in a preview mode to expose an area beyond the edge as would have been obvious in view of Inside Out, and then releases contact with the screen, the view "snaps" back to that stored in the undo stack, making the edge of the document appear to be elastically attached to the edge of the screen display. Furthermore, Robbins teaches navigating from one sector (preview state) to another sector (saved state) of a map (electronic document). Robbins teaches that the transition can be animated by elastically attaching the map to the edges of the frame or screen: When navigating from sector 5 to sector 2 (e.g., from view 610 to view 620 to view 630), the map ***shrinks and stretches*** so that the aspect of the selected child view ***fills the frame or screen***. (¶ 84, emphasis added). It would have been obvious to one of ordinary skill in the art to use

Claim	Prior Art
	the elastic animation of Robbins in combination with the panning technique of Glimpse which calls for animating the panning transition between the preview state and the saved state.
17 [a] The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching the edge of the electronic document has a first associated translating distance that corresponds to a distance of movement of the object prior to reaching the edge of the electronic document;	The Glimpse article references tracking of the stylus or finger object on a touch screen (p. 1376-77) and the translation distance prior to reaching an edge clearly can be large in comparison to dimensions of the screen.
17[b] and wherein displaying an area beyond the edge of the electronic document comprises translating the electronic document in the first direction for a second associated translating distance, wherein the second associated translating distance is less than a distance of movement of the object after reaching the edge of the electronic document.	Inside Out shows a small dimension of area beyond the edges, obviously often less than the distance the document has been translated before reaching the edge. (Fig. 27-13, p. 764). It would have been obvious to modify Glimpse by exposing a relatively small area beyond the edges as shown in Inside Out, where the movement required to expose that area is less than the distance the document was translated before reaching the edge.
19[a] A device, comprising:	The Glimpse article discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract).
19[b] a touch screen display;	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762)
19[c] one or more processors; memory; and one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors, the programs including:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Glimpse also discloses using a "Tablet PC…as our pressure sensitive input device." (p.1376)
19[d] instructions for displaying a first portion of an electronic document;	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to

Claim	Prior Art
	see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
19[e] instructions for detecting a movement of an object on or near the touch screen display;	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
19[f] instructions for translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement;	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77) In addition, the 381 patent admits that the prior art teaches this limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
19[g] instructions for displaying an area beyond an edge of the electronic	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is

Claim	Prior Art
document and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion, in response to the edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and	still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.
19[h] instructions for translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Continuing in the first direction is not an operable option for accomplishing the return. If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion prior reaching an edge obscures the previously displayed area beyond the edge. In addition, to the extent (assuming for the sake of argument) panning (translating) in a second direction is not explicitly or inherently disclosed by Glimpse, such action would have been an obvious modification of Glimpse, as modified by Inside Out, in light of the Robbins reference. One of ordinary skill in the art would have been motivated to combine Glimpse with Robbins because both solutions are directed to the problem of navigating electronic documents by panning and zooming. Robbins teaches allowing a

Claim	Prior Art
	user to "glance" at other parts of a document with the option to easily revert back to the previous location. (¶ 9). Glimpse teaches animating a transition from the preview state to the saved state. (p. 1377). Robbins also teaches animating a transition between views while the user is panning. (¶ 75, 86). In fact, Robbins teaches using a spring-loaded panning animation after a user stops glancing at a region to "snap back to the previous view." (¶ 9, 71, 75, 86). Because Glimpse calls for animating a transition, it would have been common sense to choose a simple animation that unwinds the panning and pans back from the preview state to the saved state, as more fully disclosed in Robbins. Thus, it would have been obvious to one of ordinary skill in the art to utilize the animations of the Robbins application in implementing the Glimpse technique for undo animation, as modified by Inside Out.
20[a] A computer readable storage medium having stored therein instructions, which when executed by a device with a touch screen display, cause the device to:	Glimpse discloses "multi-level input devices such as touch screens and pen-based computers." (Abstract). Inside Out discloses a touch screen Tablet PC. (Fig. 27-10, p. 762)
20[b] display a first portion of an electronic document;	Glimpse discloses allowing a user to pan to different portions of a document displayed on a multi-level touch screen, including the displaying of a first portion of an electronic document: Our technique would enable users to preview different magnification levels with light touch input before choosing to remain at the new level with a heavy touch or to return to a previous level by releasing. Similarly a user may click and drag using light input to ***pan to other portions of the document***, easily able to return to their previous position. Taking a temporary glimpse at details that are too small to see clearly (in the case of zooming) or off-screen (in the case of panning) becomes a single touch operation. (p. 1377)(emphasis added).
20[c] Detect a movement of an object on or near the touch screen display;	Glimpse teaches detecting movement of an object on the touch screen display by using a pressure sensitive touch screen to identify multi-level input, including both light touch and heavy touch: The technique we propose provides a method for editing objects with a multi-level input device such as a pressure sensitive stylus, ***pressure sensitive touch screen***, or pop-through mouse. We have used both a TabletPC and a touch

Claim	Prior Art
	sensitive DiamondTouch [3] surface as our pressure sensitive input device. (p. 1376)(emphasis added). Glimpse also teaches tracking movement of an object in contact with the pressure sensitive screen: Any multi-state input device that also ***provides tracking*** (explicitly, as in the case of the pop-through mouse's on-screen pointer, or implicitly, as in the case of a stylus or finger) can exploit this technique. (p. 1376) (emphasis added). Inside Out discloses stylus input to a touch screen Tablet PC. (Fig. 27-10, p. 762)
20[d] translate the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion, in response to detecting the movement;	Glimpse discloses panning to different portions of an electronic document using a progression of light touch to heavy touch and back again. Glimpse teaches that if a light touch is applied, a preview of a second portion of the document is displayed to the user. If the user then applies a heavy touch, the view on the screen when the heavy touch is applied becomes the new saved location. The user then returns to applying a light touch to continue previewing other portions of the document without ever breaking contact with the screen. (p. 1376-77) In addition, the 381 patent admits that the prior art teaches this limitation. The 381 patent teaches that only a portion of a large electronic document can be visible on the small screen of a portable electronic device at a given time. A user can pan electronic documents to display other portions on the screen: As a result of the small size of display screens on portable electronic devices and the potentially large size of electronic files, frequently only a portion of a list or of an electronic document of interest to a user can be displayed on the screen at a given time. Users thus will frequently need to scroll displayed lists or to translate displayed electronic documents. (Col. 2, ln 14-21).
20[e] display an area beyond an edge of the electronic document and display a third portion of the electronic document, wherein the third portion is smaller than the first portion, if the edge of the electronic document is reached while translating the electronic document in the first direction while the object is still	Glimpse discloses a user panning to display various portions of an electronic document, including a second portion, while the object is still detected on the screen with at least a light touch. (p. 1376-77). To the extent displaying an area beyond the edge of a document is not explicitly or inherently disclosed by Glimpse, doing so would have been obvious to one skilled in the art in light of Glimpse, either alone, or in combination with Inside Out, which teaches continuing

Claim	Prior Art
detected on or near the touch screen display;	translation of an electronic document when an edge is reached until an area beyond the edge is displayed on a Tablet PC. (p. 764). One of ordinary skill in the art would have been motivated to combine Glimpse with Inside Out's teaching of Microsoft Word 2003 functions because Glimpse is directed to navigating, exploring, and editing electronic documents on a Tablet PC. Glimpse teaches using a Tablet PC "as our pressure sensitive input device." (p. 1377). The Inside Out reference explains that Tablet PCs can run Microsoft Word 2003. Furthermore, it would have been common sense at the time to use a Microsoft Word 2003 document on a Tablet PC, and therefore, in combination with Glimpse.
20[f] translate the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen display.	Glimpse teaches that when contact is broken with the screen, the screen automatically pans back to the saved state of the document, which is a fourth portion that is different than the first portion. Combination of Inside Out with Glimpse as described above further includes panning beyond an edge. It would have been obvious to return by panning the document in a second direction because Glimpse teaches "animating this undo graphically" to return to the previous view. (p. 1377). Continuing in the first direction is not an operable option for accomplishing the return. If a Glimpse user has applied a heavy touch while panning around the document, the saved system undo stack (default state) is the location where the user applied the heavy touch (second portion), not the location where the user began panning with a light touch (first portion). Therefore, the restored view (fourth portion) is the same as the second portion and different from the first portion. (p. 1376-77). The view of the second portion prior reaching an edge obscures the previously displayed area beyond the edge. In addition, to the extent (assuming for the sake of argument) panning (translating) in a second direction is not explicitly or inherently disclosed by Glimpse, such action would have been an obvious modification of Glimpse, as modified by Inside Out, in light of the Robbins reference. One of ordinary skill in the art would have been motivated to combine Glimpse with Robbins because both solutions are directed to the problem of navigating electronic documents by panning and zooming. Robbins teaches allowing a user to "glance" at other parts of a document with the option to easily revert back to the previous location. (¶ 9). Glimpse teaches animating a transition from the preview state to the saved state. (p. 1377). Robbins also teaches animating a transition between views while the user is panning. (¶ 75, 86). In fact, Robbins teaches using a spring-loaded panning animation

Claim	Prior Art
	after a user stops glancing at a region to "snap back to the previous view." (¶ 9, 71, 75, 86). Because Glimpse calls for animating a transition, it would have been common sense to choose a simple animation that unwinds the panning and pans back from the preview state to the saved state, as more fully disclosed in Robbins. Thus, it would have been obvious to one of ordinary skill in the art to utilize the animations of the Robbins application in implementing the Glimpse technique for undo animation, as modified by Inside Out.

The Glimpse article discloses all of the elements of claim 1 of the '381 patent implemented on a touch screen device except displaying an area beyond an edge of the document in response to reaching the edge. Inside Out discloses displaying an area beyond an edge of the document in response to reaching the edge, on a touch screen device. Glimpse teaches animating a return to a previous view on releasing a pointing device or finger from the screen. The Robbins application teaches translating from and, upon releasing a pointing device or finger from the screen, returning to a previous view using a "spring-loaded animation." It would have been obvious in light of Inside Out to display an area beyond the edge of a document when practicing the method of Glimpse upon reaching the edge, and, upon releasing contact with the screen, to retrace the initial translating motion in the manner of a retracting spring, in light of the Robbins application. At the end of such motions, according to a mode of operation of the Glimpse method described above, the area beyond the edge of the document would no longer be displayed.

The result of applying the described teaching from Inside Out to the Glimpse article method was predictable, namely, on translating and reaching an edge, an area

beyond the edge would be displayed until the user or the system translated the document to display a portion away from the edge. The result of applying Robbins' spring-loaded return function to the method of Glimpse as modified by Inside Out was predictable, namely, to cause the animation to retrace the previous route of translation in a first direction, just as a "spring" retracts in a second direction opposite to its initial route.

One skilled in the art would have found a clear motivation to modify the Glimpse article with the teachings of Inside Out and the Robbins application, as both Glimpse and Inside Out disclosing using their navigating and display techniques on Tablet PCs and Robbins discloses using its techniques on a portable device with a touch screen or touch pad sensitive to a pointing device. Therefore, it would have been obvious to combine these three references as described to obtain the invention claimed in claim 1.

Independent claims 19 and 20 are device and computer readable medium versions of claim 1, and both read on the combination of the Glimpse article in view of the Robbins application and Inside Out according the same analysis applied above in connection with claim 1. Dependent claims 2-11, 13-16, and 17 would have been obvious for the reasons stated in the above claims chart for this second ground of rejection.

(c) ***Third Ground for Rejection*****:** Claims 5, 9, 11, 12, 15, and 18 would have been obvious over the Glimpse article in view of Inside Out, the Robbins application, and the Zimmerman patent.

By comparing the content of the Glimpse article, Inside Out, the Robbins application, and the Zimmerman patent to claims 5, 9, 11, 12, 15, and 18 of the '381

patent, it will become clear that the claimed subject matter would have been obvious to one of ordinary skill in the art.

Claim	Prior Art
5. The computer-implemented method of claim 1, wherein the first direction is a vertical direction, a horizontal direction, or a diagonal direction.	The analysis for claims 1 and 5 provided in the Second Ground for Rejection above is incorporated here. The Zimmerman patent further teaches translation in a vertical direction by indicating that: "Electronic image displays of lists that extend beyond the vertical display dimension of the display screen are displaced in the vertical direction by touching the screen with a finger and then moving the finger in the desired direction on the screen." (Abstract). It would have been obvious to translate documents vertically in the Glimpse method in view of the Zimmerman patent.
9. The computer-implemented method of claim 1, wherein the electronic document includes a list of items.	The analysis for claims 1 and 9 provided in the Second Ground for Rejection above is incorporated here. The Zimmerman patent further teaches translation in a vertical direction by indicating that: "Electronic image displays of lists that extend beyond the vertical display dimension of the display screen are displaced in the vertical direction by touching the screen with a finger and then moving the finger in the desired direction on the screen." (Abstract). It would have been obvious to translate documents including displays of lists vertically in the Glimpse method in view of the Zimmerman patent.
11. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching an edge of the document has an associated speed of translation that corresponds to a speed of movement of the object.	The analysis for claims 1 and 11provided in the Second Ground for Rejection above is incorporated here. The Zimmerman patent teaches that in a natural manner, the initial speed of displacement of the displayed image corresponds to the speed of motion of the finger along the screen. It would have been obvious to modify the combined teachings as described to provide translation speed corresponding to movement of the pointing object, in view of Zimmerman. (Abstract; col. 3, ln. 54-57).
12. The computer-implemented method of claim 1, wherein translating in the first direction is in accordance with a simulation of an equation of motion having friction.	The analysis for claim 1 provided in the Second Ground for Rejection above is incorporated here. Zimmerman discloses translating a list (electronic document) in a first direction while the user maintains contact with the screen, and under some conditions, applies damping to the translation speed: Electronic image displays of lists that extend beyond the vertical display dimension of the display screen are displaced in the vertical direction by touching the screen with a finger and then moving the finger in the desired direction on the screen. In a natural manner the initial speed of displacement of the displayed image corresponds to the speed of motion of the finger along the screen. When the user's finger is disengaged from the screen, the system senses the disengagement and thereafter allows the vertical

Claim	Prior Art
	displacement speed of the image to decrease at a controlled rate. (Abstract). Decreasing a displacement speed at a controlled rate would have been understood to be a damped motion, which the '854 patent treats as the same as following an equation of motion having friction. It would have been obvious to modify the combined teachings as described to provide translation speed in the first direction corresponding to simulation of an equation of motion having friction, in view of Zimmerman. (See col. 3, ln. 54 – col. 4, ln. 37).
15. The computer-implemented method of claim 1, wherein translating the document in the second direction is a damped motion.	The analysis for claims 1 and 15 provided in the Second Ground for Rejection above is incorporated here. Zimmerman discloses translating a list (electronic document) in a first direction while the user maintains contact with the screen. When the user breaks contact, the list continues translating but the speed slowly decreases until it reaches zero: Electronic image displays of lists that extend beyond the vertical display dimension of the display screen are displaced in the vertical direction by touching the screen with a finger and then moving the finger in the desired direction on the screen. In a natural manner the initial speed of displacement of the displayed image corresponds to the speed of motion of the finger along the screen. When the user's finger is disengaged from the screen, the system senses the disengagement and thereafter allows the vertical displacement speed of the image to decrease at a controlled rate. (Abstract). (See col. 3, ln. 54 – col. 4, ln. 37). Decreasing a displacement speed at a controlled rate would have been understood to be a damped motion. It would have been obvious to modify the combined teachings as described to provide a damped translation speed in the second direction, in view of Zimmerman and Robbins.
18. The computer-implemented method of claim 1, wherein translating in the first direction prior to reaching the edge of the electronic document has a first associated translating speed that corresponds to a speed of movement of the object, and wherein displaying an area beyond the edge of the electronic document comprises translating the electronic document in the first direction at a second associated translating speed,	The analysis for claim 1 provided in the Second Ground for Rejection above is incorporated here. Zimmerman discloses panning a list (electronic document) in a first direction until a specified event occurs. In Zimmerman, the specified event is the user breaking contact with the screen. As long as this event does not occur, the list continues panning at a first speed. When the user breaks contact, the list continues panning but the speed is decreased. (Abstract) It would have been obvious to one skilled in the art to apply the

Claim	Prior Art
wherein the second associated translating speed is slower than the first associated translating speed.	speed decrease of Zimmerman after other kinds of events occur, such as reaching the edge of an electronic document. After reaching an edge, the user would naturally tend to break contact with the screen, resulting in a lower translation speed in a Zimmerman method. (Col. 3, ln. 54 – col. 4, ln. 37). It would have been obvious to one of ordinary skill in the art to modify the combined teachings as described to provide a damped (lower) translation speed during the display of an area beyond the edge of the electronic document, in view of Zimmerman and Robbins.

Claims 5, 11, 12, 15, and 18 depend from Claim 1, which in the First and Second Grounds for Rejection has been shown to read on obvious combinations of prior art not previously considered by the Office.

Claims 5 and 9 are met, respectively, by document movement in the first direction being vertical, and by the document being a list, which are taught by the Zimmerman patent explicitly.

Claim 11 requires a speed of translation prior to reaching an edge of the document that corresponds to the speed of a finger or other object along the screen, which is taught by the Zimmerman patent explicitly.

Claim 12 requires that movement in the first direction is in accordance with a simulation of an equation of motion having friction. The Zimmerman patent teaches damping translating motions. The result of applying this known technique to translation of documents according to the Glimpse article would have been predictable in that a damping effect would have been applied to the translation.

Claim 15 requires damped motion in the second direction, as the area beyond the edge becomes no longer displayed. Zimmerman's damped motion provides a smooth

animation that was predictably applicable to the return animation techniques taught by the other references.

Claim 18 requires moving the document more slowly as the area beyond the edge is displayed. Zimmerman teaches damping the translating motion at the end of a translation. It would have been predictably obvious to apply such damping when displaying an area beyond the edge of a document according to the combination of Glimpse and Inside Out.

Like the other references, Zimmerman relates to translation of electronic documents displayed on a touch screen. It would have been obvious to modify the combination of the Glimpse article method modified by teachings of Inside Out and the Robbins application in accordance with the foregoing teachings of the Zimmerman patent to meet dependent claims 5, 11, 12, 15, and 18.

III. Conclusion

The Glimpse article, Inside Out, and the Robbins application, were not previously considered, and they are not cumulative with the references previously considered. Consideration of obvious combinations of these references in accordance with the proposed Grounds for Rejection, and, for some dependent claims, in view of the Zimmerman patent, leads to the conclusion that these references create substantial new questions of patentability for claims 1-20 of the '381 patent. Third party requester further submits that claims 1-20 must be rejected as unpatentable.

Respectfully submitted,

/Scott E. Brient/

Scott E. Brient
Reg. No. 44,561

Customer No. 93485
Brient Intellectual Property Law, LLC
2300 Lakeview Parkway, Suite 700
Alpharetta, GA 30009
Phone: (678) 916-3828
Fax: (678) 835-9310

ELECTRONICALLY FILED USING THE EFS-WEB ELECTRONIC FILING SYSTEM OF THE UNITED STATES PATENT & TRADEMARK OFFICE ON April 28, 2010.

EXHIBIT 9

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/010,963	04/28/2010	7469381	0919/01028	8062

61725 7590 07/14/2010

Morgan Lewis & Bockius LLP/ AI
2 Palo Alto Square
3000 El Camino Real, Suite 700
Palo Alto, CA 94306

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 07/14/2010

Please find below and/or attached an Office communication concerning this application or proceeding.

Order Granting / Denying Request For Ex Parte Reexamination	Control No. 90/010,963	Patent Under Reexamination 7469381
	Examiner Christina Y. Leung	Art Unit 3992

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

The request for *ex parte* reexamination filed *28 April 2010* has been considered and a determination has been made. An identification of the claims, the references relied upon, and the rationale supporting the determination are attached.

Attachments: a)☐ PTO-892, b)☒ PTO/SB/08, c)☐ Other: _____

1. ☒ The request for *ex parte* reexamination is GRANTED.

RESPONSE TIMES ARE SET AS FOLLOWS:

For Patent Owner's Statement (Optional): TWO MONTHS from the mailing date of this communication (37 CFR 1.530 (b)). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).**

For Requester's Reply (optional): TWO MONTHS from the **date of service** of any timely filed Patent Owner's Statement (37 CFR 1.535). **NO EXTENSION OF THIS TIME PERIOD IS PERMITTED.** If Patent Owner does not file a timely statement under 37 CFR 1.530(b), then no reply by requester is permitted.

2. ☐ The request for *ex parte* reexamination is DENIED.

This decision is not appealable (35 U.S.C. 303(c)). Requester may seek review by petition to the Commissioner under 37 CFR 1.181 within ONE MONTH from the mailing date of this communication (37 CFR 1.515(c)). **EXTENSION OF TIME TO FILE SUCH A PETITION UNDER 37 CFR 1.181 ARE AVAILABLE ONLY BY PETITION TO SUSPEND OR WAIVE THE REGULATIONS UNDER 37 CFR 1.183.**

In due course, a refund under 37 CFR 1.26 (c) will be made to requester:

a) ☐ by Treasury check or,

b) ☐ by credit to Deposit Account No. _____, or

c) ☐ by credit to a credit card account, unless otherwise notified (35 U.S.C. 303(c)).

/Christina Y. Leung/ Primary Examiner, Art Unit 3992		

cc:Requester (if third party requester)

DECISION GRANTING EX PARTE REEXAMINATION

Decision on the Request

The present request for *ex parte* reexamination raises a substantial new question of patentability with respect to **claims 1-20** of United States Patent 7,469,381 to **Ording**.

References Cited in the Request

Glimpse (Forlines et al., "Glimpse: A Novel Input Model for Multi-Level Devices," Conference on Human Factors in Computing Systems CHI '05, Association for Computing Machinery, 2005, pp. 1375-1378)

Inside Out (Millhollon et al., "Microsoft Office Word 2003 Inside Out," Microsoft Press, 2004, pp. 13-16, 93, 762-765, 802-804)

Robbins (US 2005/0195154 A1)

Zimmerman (US 6,690,387 A)

Issues Raised by the Request

Issue 1

The request alleges that Glimpse in combination with Inside Out raises a substantial new question of patentability with respect to claims 1-11, 13, 14, 16, 17, 19, and 20.

Issue 2

The request alleges that Glimpse in combination with Inside Out and Robbins raises a substantial new question of patentability with respect to claims 1-11, 13-17, 19, and 20.

Issue 3

The request alleges that Glimpse in combination with Inside Out, Robbins, and Zimmerman raises a substantial new question of patentability with respect to claims 5, 9, 11, 12, 15, and 18.

The Ording Patent

The Ording patent is generally directed to method and computer-readable instructions for displaying an electronic document on a touch screen display in response to movement of an object on or near the touch screen. Claim 1 is representative:

1. A computer-implemented method, comprising:

at a device with a touch screen display:

displaying a first portion of an electronic document;

detecting a movement of an object on or near the touch screen display;

in response to detecting the movement, translating the electronic document displayed on the touch screen display in a first direction to display a second portion of the electronic document, wherein the second portion is different from the first portion;

in response to an edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display:

displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion; and

in response to detecting that the object is no longer on or near the touch screen display, translating the electronic document in a second direction until the area beyond the edge of the

electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion.

Prosecution History

Claims 1-20 are the current claims of the Ording patent, which issued 23 December 2008 from application 11/956,969 filed 14 December 2007. Application 11/956,969 claims priority to provisional applications 60/937,993 filed 29 June 2007; 60/946,971 filed 28 June 2007; 60/945,858 filed 22 June 2007; 60/879,469 filed 08 January 2007; 60/883,801 filed 07 January 2007; and 60/879,253 filed 07 January 2007.

14 December 2007: Applicant originally filed claims 1-20.

18 April 2008: The Office granted Applicant's petition for accelerated examination.

30 April 2008: Applicant filed an examination support document, citing Zimmerman (US 6,690,387 A), Kwatinetz (US 5,495,566 A), Pallakoff (US 2005/0012723 A1), and Miller ("Personal Java Application Environment," 1999) as references deemed most closely related to the claims.

02 June 2008: Examiner initiated an interview with Applicant and discussed Zimmerman, Microsoft Word screenshots, and Collins (US 2008/0104544 A1). Applicant agreed to amend the independent claims to include "in response to detecting that the object is no longer on or near the touch screen display, translating the document in a second direction until the area beyond the edge of the document is no longer displayed."

04 August 2008: Examiner initiated an interview with Applicant and discussed Photo Mesa screenshots and Jaeger (US 2004/0027398 A1). Applicant proposed amended claims, and Examiner agreed that they were allowable over the cited art.

29 October 2008: Examiner amended claims 1, 2, 19, and 20 by Examiner's amendment and allowed claims 1-20, noting that:

"In regards to the independent claims 1, 19 and 20, the prior art found does not teach in response to an edge of the electronic document being reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display: displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion; and in response to detecting that the object is no longer detected on or near the touch screen display, translating the electronic document in a second direction until the area beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion; in combination with all of the other claim limitations."

Detailed Analysis

Claims 1-20 will be reexamined. In view of the prosecution history, a substantial new question of patentability is raised by the evaluation of a prior art reference (or a combination of prior art references) that teaches the features and limitations added to independent claims 1, 19, and 20 in the 29 October 2008 Examiner's amendment and highlighted in Examiner's reasons for allowance. Specifically, these limitations include displaying an area beyond the edge of the document, and displaying a third portion of the electronic document, wherein the third portion is smaller than the first portion, if the edge of the electronic document is reached while translating the electronic document in the first direction while the object is still detected on or near the touch screen display; and translating the electronic document in a second direction until the area

beyond the edge of the electronic document is no longer displayed to display a fourth portion of the electronic document, wherein the fourth portion is different from the first portion, in response to detecting that the object is no longer on or near the touch screen.

Issues 1-3

Glimpse, Inside Out, and Robbins are new prior art. Zimmerman was previously cited by the examiner but was not considered in combination with Glimpse, Inside Out, and Robbins. Glimpse teaches, among other things, translating an electronic document on a touch screen in a first direction to display a second portion of the document different from the first portion when an object is on or near the touch screen. Glimpse further teaches translating the document in a second direction to display another portion of the document, different from the first portion, when the object is no longer on or near the touch screen (i.e., Glimpse teaches panning back to a saved state when contact is broken with the screen; page 1377). Inside Out teaches, among other things, displaying an area beyond the edge of an electronic document and displaying a portion of the document smaller than a first portion if the edge of the document is reached while translating the document in a first direction (page 764).

Since these teachings are directly related to subject matter considered the basis for allowability of claims 1-20, a reasonable examiner would consider evaluation of Glimpse in combination with Inside Out important in determining the patentability of the claims. Therefore, Glimpse in combination with Inside Out only, or with Inside Out and further references, raises a substantial new question of patentability with respect to claims 1-20.

Conclusion

Extensions of time under 37 CFR 1.136(a) will not be permitted in these proceedings because the provisions of 37 CFR 1.136 apply only to "an applicant" and not to parties in a reexamination proceeding. Additionally, 35 U.S.C. 305 requires that *ex parte* reexamination proceedings "will be conducted with special dispatch" (37 CFR 1.550(a)). Extensions of time in *ex parte* reexamination proceedings are provided for in 37 CFR 1.550(c).

The patent owner is reminded of the continuing responsibility under 37 CFR 1.565(a) to apprise the Office of any litigation activity, or other prior or concurrent proceeding, involving Patent No. 7,469,381 throughout the course of this reexamination proceeding. The third party requester is also reminded of the ability to similarly apprise the Office of any such activity or proceeding throughout the course of this reexamination proceeding. See MPEP §§ 2207, 2282 and 2286.

All correspondence relating to this *ex parte* reexamination proceeding should be directed:

By mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By fax to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Any inquiry concerning this communication should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

/Christina Y. Leung/

Primary Examiner, Art Unit 3992

/D. M. H./
Primary Examiner, Art Unit 3992
ESIC

Substitute for form 1449/PTO
(Revised 07/2007)

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet	1	of	1

Complete if Known	
Application Number	11/956,969
Filing Date	December 14, 2007
Patent Number	7,469,381 (Exhibit A)
Issue Date	December 23, 2008
First Named Inventor	Ording
Art Unit	2174
Examiner Name	B. Pesin
Attorney Docket Number	0919/01028

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.	Document Number Number - Kind Code (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages of Relevant Figures Appear
/CL/	1	US-2005/0195154 A1 (Exhibit D)	09-08-2005	Robbins et al.	
/CL/	2	US-6,690,387 B2 (Exhibit E)	02-10-2004	Zimmerman et al.	

OTHER DOCUMENTS

Examiner Initials*	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s) , volume-issue number(s), publisher, city and/or country where published.	English Language Translation Attached
/CL/	3	FORLINES et al., Glimpse: A Novel Input Model for Multi-Level Devices, April 2005, 6 pages total, Mitsubishi Electric Research Laboratories, Cambridge, MA (Exhibit B)	
/CL/	4	MILLHOLLON et al., Microsoft Office Word 2003 Inside Out, 2003, Microsoft Press, Redmond, Washington, pages 93, 762-765. (Exhibit C)	

Examiner Signature	/Christina Leung/	Date Considered	07/08/2010

*Examiner: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.